PCI Bus CPD Board Series

HPCI-CPD578

User's Manual

Individual Board Information

Multifunctional High-speed NC Boards with
Circular Interpolation, Linear Interpolation, and Positioning

Hivertec Inc.

http://www. hivertec. co. jp/

This manual applies to the following Boards in the CPD Series.

CPD570 Series: PCI Bus

HPCI-CPD578

It is prohibited to reprint or copy all or any part of this manual or programs without prior written permission.
The contents of this manual are subject to change without notice to enable improvements to be made.
Please notify your sale representative for this product if you find any problems with the contents of this manual.

Windows 95, Windows 98, Windows NT 4.0, Windows 2000, WindowsXP, Visual C++, Visual Basic,
and Microsoft C/C++ are registered trademarks of the Microsoft Corporation in the USA and other countries.
Other company and product names are the trademarks or registered trademarks of the respective companies.

Hivertec Inc.
4-8-1 Sumida-Ku Ryogoku,
130-0026 Tokyo Japan
TEL. +81-3-3846-3801
FAX. +81-3-3846-3773
e-mail. sales@hivertec.co.jp

Revision 3.2e 04/04/22 released
Copyright Hivertec Inc.

Extent of Warranty

1. The product warranty is valid for a period of one year from the date of purchase. If a defect is acknowledged by Hivertec within the period of warranty, Hivertec will repair or replace the product upon return of the product to Hivertec.

2. Hivertec is not responsible beyond the purchase price of the product for any damages or loss of profit, direct, indirect, or secondary, caused by application, delivery, or failure of a Hivertec product either within or outside of the period of warranty.

Limitations to Liability

1. Hivertec is not responsible for any damages resulting from product installation, connections, settings, or operation that do not follow the contents of this manual.

2. This product uses semiconductor devices manufactured for general electronics equipment, such as machine tools, instrumentation, FA devices, OA devices, and communications equipment. They are not designed, conceived, approved for, or warranted for application in devices for which faulty operation or failure will have a direct affect on human life or result in personal injury or damage to property. The safety, quality, and performance of the product are not guaranteed explicitly or implicitly beyond those given in this manual or related catalogs.

3. Hivertec is not responsible for any damages resulting from modifications or repairs made to the product without the approval of Hivertec either within or outside of the period of warranty.

4. The contents of this manual do not guarantee or grant rights to patents, copyright, trademark rights, or any other rights to the intellectual property of Hivertec or any third party. Hivertec is not responsible for any problems that may occur concerning the rights to intellectual property of third parties resulting from the application of information provided in this manual.

Important Safety Instructions

Thank you for choosing a Hivertec NC Board. This manual contains information that is important for the safe and reliable operation of the NC Board. Read this section and understand the information contained before attempting to use the NC Board.
Furthermore, save this manual and store it in an easily accessible location near the installed NC Board, so that it can be referenced when necessary.

Safety Precautions
Always read this manual and any attached documents completely before attempting to use the product. Be sure that you understand the information provided and are using the product correctly. Do not use the product before having a complete understand of the product, product safety information, and precautions. In this manual, safety precautions are classified as either Warnings or Cautions..

Warning	Indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.
Caution	Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury, or property damage. .

1. Authorized For

Caution

The product and this manual are designed for those with the following knowledge.

- **A basic knowledge of installing and wiring expansion boards.**
- **A basic knowledge of electronic control devices and personal computers.**

2. Applicable Bus

Warning

The boards (products) described in this manual are applicable to PCI Local Bus Specification Rev.2. 1 (+5 V specifications). Do not use them in any environment where this bus is not operating.

3. Environmental Conditions

Warning

Store and use the board only under the following environmental conditions.

- Ambient operating temperature 0 to 50 °C
- Ambient operating humidity: 20% to 85% (with no condensation)
- Ambient storage temperature: -15 to 75 °C
- Ambient storage humidity: 10% to 90% (with no condensation)
- Atmosphere: No corrosive gas, inflammable gas, oil mist, or dust
- Altitude: 3,000 m above sea level max.
  (Lower upper level of temperature ranges by 2 °C for every 300 m.)

4. Transportation and Installation

Warning

Touch a metal object to discharge static electricity from your body before touching the board. Static electricity may cause board failure.

Do not wrap the board in any wrapping material that easily carries a static charge, such as bubble wrap. Static electricity may cause board failure.

Do not touch the edge connector on the board. Contamination on the edge connector may cause operating errors.

Do not place any heavy object on the board. Heavy object may damage components on the boards, possibly causing board failure.

Set the jumpers on the board before mounting it to the computer. If the jumpers are set with the power supply turned ON, the settings may not be recognized correctly possibly causing operating errors.

Set the board jumpers correctly. Incorrect settings may cause operating errors.

Always turn OFF the power supply to the computer and disconnect the power supply cable before installing the board. Installing the board without removing the power supply cable may cause board failure. Furthermore, the device may exhibit unexpected operation.

When installing the board in the computer, be sure the board is straight to the connector in the computer and press in the board so that the gold-plated edge connector is deeply inserted into the PCI connector. If the board is inserted at an angle or not inserted deeply enough, poor contact may cause operating error or board failure.

When installing the board in the computer, use mounting brackets and mounting screws to secure it firmly. Improper securing of the board may cause operating errors.

Caution

Do not drop the board or handle it roughly. Vibration or shock may cause board failure.

Do not touch the solder surface of the board with your hands. Sharp points on the components may cause injury.

5. Wiring

Warning

Always turn OFF the power supply to the computer and disconnect the power supply cable before wiring connectors to exterior lines or before connecting or disconnecting such connectors. Performing this work without removing the power supply cable may cause board failure. Furthermore, the device may exhibit unexpected operation.

When wiring connectors to exterior lines, check connector signal tables carefully and wire all lines correctly. Incorrect wiring may cause board failure or burning.

Always keep external power supply within ratings. Using a power supply that is not within ratings may cause board failure, burning, or operating errors.

Always keep circuits connected to the I/O circuits within the rated currents and voltages. Using circuits that are not within ratings may cause board failure, burning, or operating errors.

Use the recommended connectors for external lines. Using any other connector may cause operating errors because of faulty contact.

Always lock connectors for external lines. A connector that is not locked may come loose, possibly causing operating errors because of faulty contact.

Do not pull on or place heavy objects on cables for external wiring. The connector may come loose and faulty contact may cause operating errors.

Separate cables for external wiring as far as possible from AC power cables, motor cables, or other cables that generate excessive noise. Noise may cause operating errors.

6. Trial Operation and Adjustment

Warning

Always debug the program completely before using the board to drive a device. Any error in the program may cause unexpected operation.

When using sample programs provided with the product to operate a device, always start at low speed and be sure that settings match the mechanical system before attempting operation. Operating with settings that do not match the mechanical system may result in unexpected operation.

7. Disposal

Warning
Abide by all applicable laws and ordinances when disposing of a board.

Manual Configuration

The following manuals are provided with CPD-series Boards.

1. User's Manual: Individual Board Information (this manual)

This manual provides the following information for individual CPD Boards.

(1) Hardware information
(2) Installation procedures for accessory software
(3) Operating procedures for sample software
(4) Operating procedures for the Trial Tour
(5) Other specific information for individual Board

2. CPD Board Series User's Manual: Software

This manual describes the following software information for the CPD Board Series.

(1) Library functions (library function level 1: VC++, VB, and DOS)
(2) Driver functions (device driver interface library: VC++, VB, and DOS)

3. CPD Board Series User's Manual: Common Information

This manual provides information common to all of the CPD Board Series in tutorial style.

(1) Basic operating methods for CPD Boards (including samples using library functions)
(2) Descriptions of CPD Board applications (samples of using library functions as reference required for a broader range of application)
(3) Descriptions of registers based on the PCL6045
(4) Other information

Contents

1. Introduction 1
1.1 About This Manual 1
1.2 Provided Software 1
1.3 Axis Names 1
2. Accessories (Sold Separately) 2
3. Hardware 3
3.1 Block diagram 3
3.2 Port Addresses 4
3.2.1 Board Addresses 4
3.2.2 Option Ports 5
3.2.3 Board I/O and Device Drivers 7
3.2.4 PCI configuration registers 7
3.3 Board Settings 8
3.3.1 CPD578 Board 8
3.4 Servo and Machine Interfaces 10
3.4.1 Command Pulse Output and Driver Connections 10
3.4.2 Axis Sensor and Servo Interface Input Circuits 11
3.4.3 Encoder Input Circuits 12
3.4.4 Servo Interface Output Circuits 13
3.4.5 Simultaneous Start/Stop and Match Signal Output 14
3.5 Connector PIN out 15
3.5.1 CPD578 connector PIN out 15
3.6 HPCI-CPD578 Specification 17
4. Functions 18
4.1 Combining Operations Between Axes 18
4.1.1 CPD578 Axis Operation Combinations 18
4.2 Interrupt Mechanisms 19
5. Software Startup Guide 20
5.1 Overview 20
5.2 Software Configuration 20
5.3 Installing and Uninstalling the Device Drivers 22
5.3.1 Installing/Uninstalling Device Drivers in a Windows Systems 22
5.3.2 Installing/Uninstalling Device Drivers in a DOS System 26
5.4 Using Two or More CPD Boards 27
5.5 Accessing the Board 28
5.5.1 Data Structure for Board (Device) Recognition 28
5.5.2 Board Access Preparatory Procedure and End Processing 28
5.5.3 Verifying Each Axis is Ready for Operation 30
5.6 Sample Programs for Windows Systems 31
5.6.1 Executing the Sample Program 31
5.6.2 Sample Program Operations 32
5.7 ”Let’s try to move” Program for Windows Systems 37
5.7.1 ”Let’s try to move” Operation Window 38
5.7.2 ”Let’s try to move” Settings Window 40
5.8 Sample Programs for a DOS System 41
5.8.1 Configuration of the Sample Program 41
5.8.2 Starting the Sample Program 41
5.8.3 Sample Program Operations 44

Contents of Tables and Figures

1. Introduction 1
2. Accessories (Sold Separately) 2
Table 2-1 Accessories (Connector Boards and Cables) 2
Figure2-1 Accessories(ConnectorBoard and Cables) 2
3. Hardware 3
Figure 3. 1-1 BLOCK DIAGRAM of CPD578 3
Table 3. 2-1 Board Addresses 4
Table 3. 2-2 Board I/O and Driver Functions 7
Table 3. 2-3 PCI configuration register 7
Figure 3. 3-1 Jumper Pin Locations on the CPD578 Board 8
Table 3. 3-1 Board ID Setting 8
Figure 3. 3-2 Termination setting of encoder A-phase and B-phase 8
Figure 3. 3-3 Encoder Z-phase input selection 8
Figure 3. 3-4 “E101” jumper for setting of ‘emergency stop’ 9
Table 3. 4-1 Command Pulse Output Circuits 10
Table 3. 4-2 Axis Sensor and Servo Interface Input Circuits 11
Table 3. 4-3 Encoder Input Circuit 12
Table 3. 4-4 Servo Interface Output Circuits 13
Table 3. 4-5 J1 – J3:Match signal and J3:Simultance Start/Stop Output Circuits 14
Figure 3. 5-1 Front panel 15
Table 3. 5-1 CPD578 J1 CONNECTOR PIN OUT (Xaxis - Uaxis) 15
Table 3. 5-2 CPD578 J2 CONNECTOR PIN OUT (Vaxis - Baxis) 16
Table 3. 5-3 CPD578 J3 CONNECTOR PIN OUT 16
Table 3. 6-1 HPCI-CPD578 Specifications 17
4. Functions 18
Table 4. 1-1 CPD578 Axis Operation Combinations 18
Figure 4. 2-1 Route of Interrupts in the CPD578 Board 19
5. Software Startup Guide 20
Figure 5. 2-1 Software Configurations 21
Figure 5. 3-1 WinXP Installations 22
Figure 5. 3-2 Win2K Installations 24
Figure 5. 3-3 WinNT Installations 24
Figure 5. 3-4 Win98 Installations 25
Figure 5. 3-5 Uninstalling the Device Drivers 26
Figure 5. 4-1 Using Multiple Boards 27
Figure 5. 6-1 Sample Program Error Messages 31
Figure 5. 6-2 The Sample Program’s Select Operation Menu 32
Figure 5. 7-1 Let’s try to move Error Messages 37
Figure 5. 7-2 Let’s try to move Operation Confirmation Window 38
Figure 5. 7-3 Let’s try to move Settings Window 40
Figure 5. 8-1 Startup Screens for the DOS Version of the Sample Program 42

1. Introduction

This user's manual provides information for the HPCI-CPD578 CPD Board that provides 8-axis interpolation and positioning. This board is in the CPD570-series PCI-bus Motion Control Boards.
Use this manual together with the *CPD Board Series User's Manual: Common Information*, which contains information common to all of the Boards in the CPD Board Series.

In the remainder of this manual, the HPCI-CPD578 is referred to as the CPD578.
And the PCL6045 series are referred to as the PCL.

1.1 About This Manual

This manual provides the following information.

- Software Startup Guide (for Windows and DOS)
  1) Installation of the provided software
  2) Explanation of the sample programs
  3) Testing operation "Trial Tour"
- Hardware Information
  1) Port address
  2) Board settings
  3) Connector allocation
  4) Servomotor interface and stepping-motor interface
  5) Machine interface
- Information on Functions Specific to the CPD578.
  1) Axis combinations
  2) Interrupts

1.2 Provided Software

The following software is provided with this Board. Refer to this software when reading this manual to facilitate understanding.

Device Drivers for Windows	There are three kinds for Windows 98, Windows NT, and Windows 2000
Library Functions for Windows (Level 1)	Provides functions necessary for basic operations. Refer to the *CPD Board Series User's Manual: Software Information* for descriptions of the functions.
Windows Sample Programs	Sample software showing usage of library functions.
"Trial Tour" for Windows	Enables minimal operations by simply connecting Board to computer. Can also be used to check connections.
Device Drivers for DOS	Request these drivers when required.

Note : In the remainder of this manual, both Windows XP Home Edition and Windows XP Professional are referred to as Windows XP.

1.3 Axis Names

When using the 8-axis CPD578, the four axes are known as the X, Y, Z, U and V, W, A, B, axes.

2. Accessories (Sold Separately)

The following Connector Boards and Cables are available as accessories to simplify connections between the CPD Board and the motor driver or machine sensors.

■ Connector Boards: The ACB-HU1004/* Boards allow each axis to be connected to motor drivers and sensors very easily through an MIL header connector.
The ACB-DX100/* Boards connect through a terminal block. These are ideal for testing.

■ Cables: There are two kinds of Cable available: the HCL-051W Connector Board Cable (standard length 2 m, other lengths available by special-order) and the HCL-051 Cable (laminated).

CPD Board	Compatible Cable	Pins	Connector Board		Remarks
HPCI-CPD578	HCL-051W	100	Connector style	ACB-HU1004/*	The asterisk at the end of the model number represents one of the following codes: DR: Right-angle connector DS: Straight connector DS(D): Half-pitch DIN Track-mounting connector
			Terminal-block style	ACB-DX100/*	

Table 2-1 Accessories (Connector Boards and Cables)

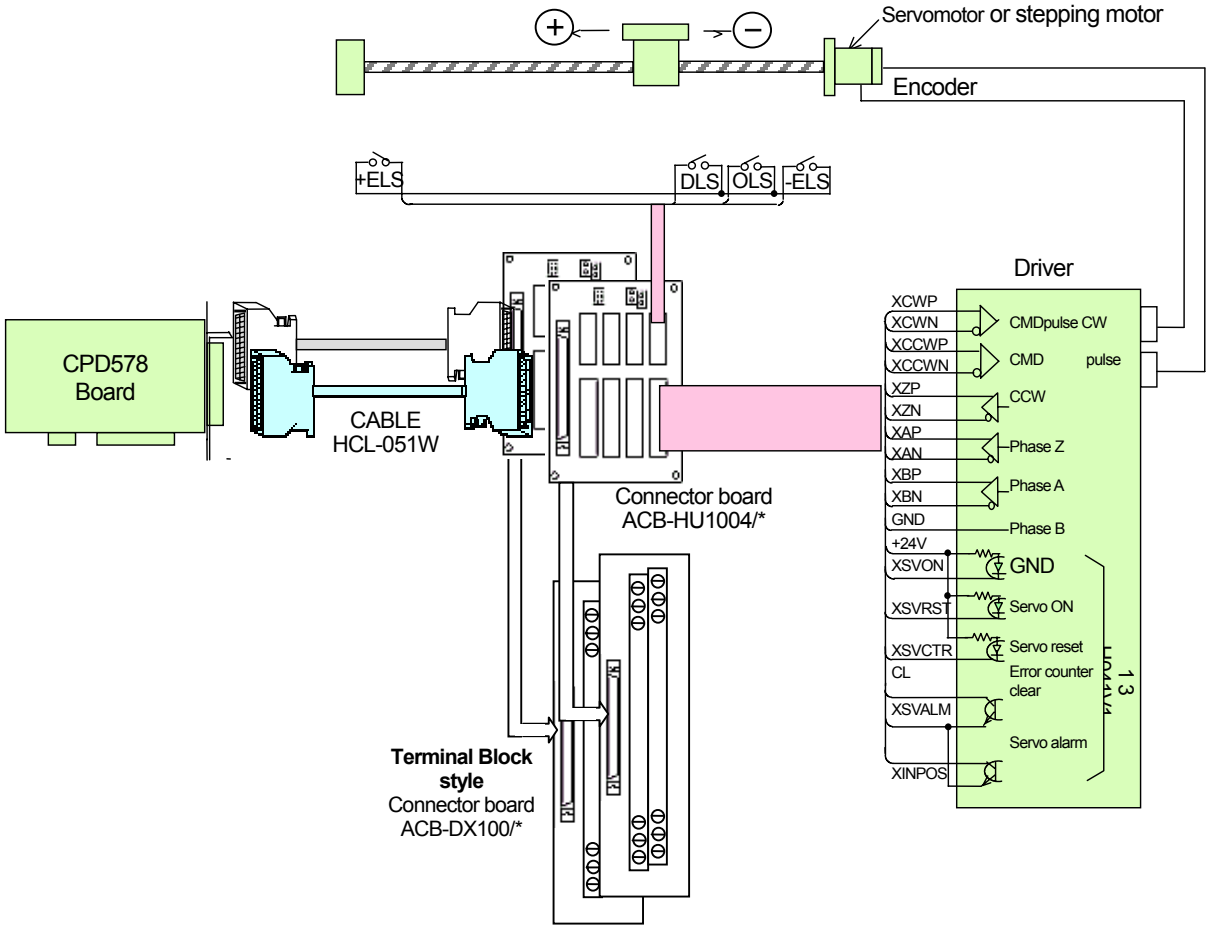
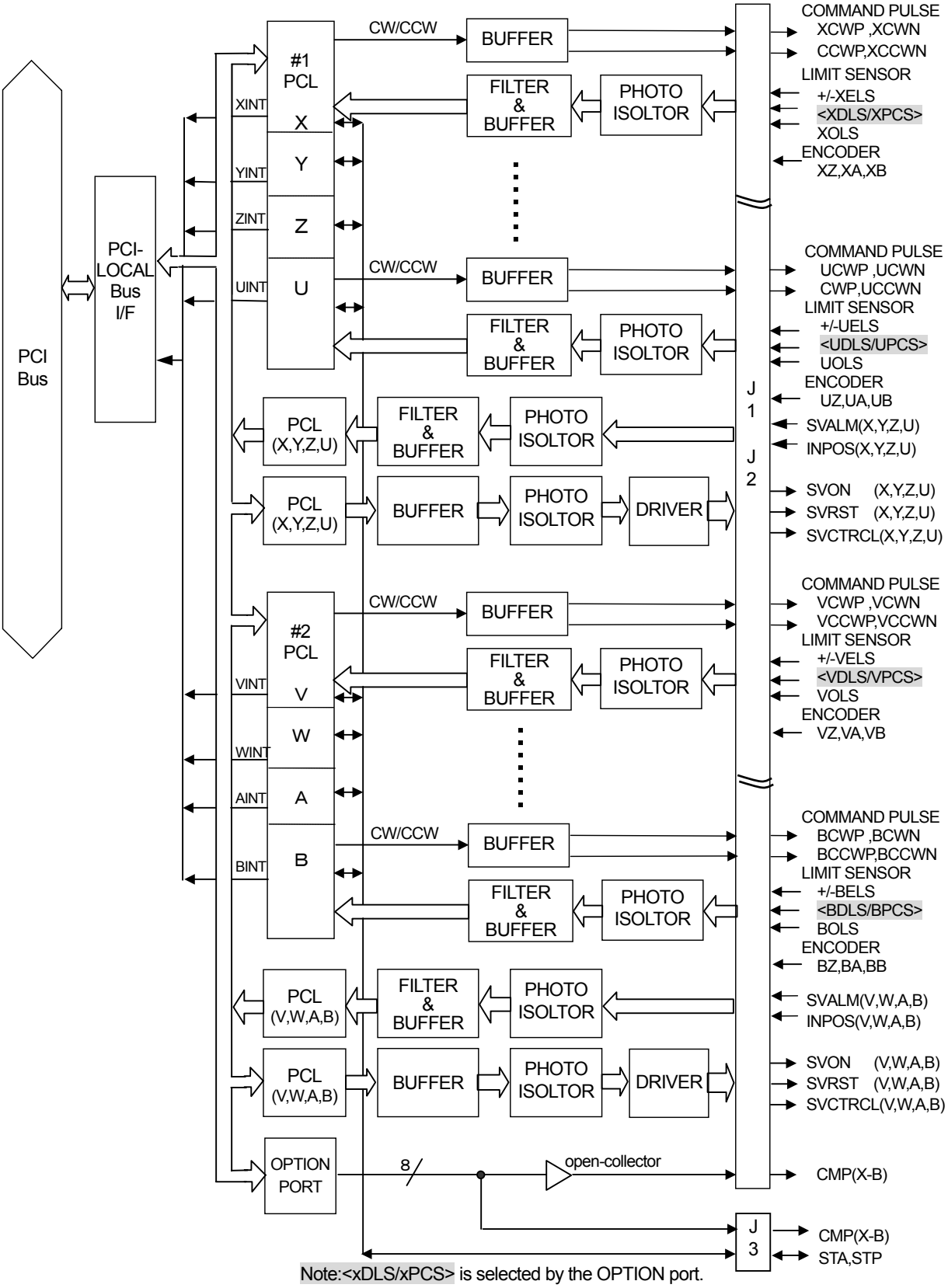


Figure2-1 Accessories(ConnectorBoard and Cables)

3. Hardware

3.1 Block diagram



Note:<xDLS/xPCS> is selected by the OPTION port.

Figure 3. 1-1 BLOCK DIAGRAM of CPD578

3.2 Port Addresses

3.2.1 Board Addresses

I/O maps are used for all ports. *Table 3. 2-1* shows the ports.
Knowledge of the following table is not required to use the provided software library (level 1) functions.
The following table is required when using the provided software driver functions.

Classification	I/O Add.	Read (INP)		Write (OUT)	
		Name	Content	Name	Content
PCL X-axis (Axis 1)	+0	MSTS	Main status	CMD	Command
	+2	SSTS	Sub-status	OTP	Not used (reserved)
	+4	BUF0	I/O buffer IN (15 to 0)	BUF0	I/O buffer OUT (15 to 0)
	+6	BUF1	I/O buffer IN (31 to 16)	BUF1	I/O buffer OUT (31 to 16)
PCL Y-axis (Axis 2)	+8	MSTS	Main status	CMD	Command
	+a	SSTS	Sub-status	OTP	Not used (reserved)
	+c	BUF0	I/O buffer IN (15 to 0)	BUF0	I/O buffer OUT (15 to 0)
	+e	BUF1	I/O buffer IN (31 to 16)	BUF1	I/O buffer OUT (31 to 16)
PCL Z-axis (Axis 3)	+10	MSTS	Main status	CMD	Command
	+12	SSTS	Sub-status	OTP	Not used (reserved)
	+14	BUF0	I/O buffer IN (15 to 0)	BUF0	I/O buffer OUT (15 to 0)
	+16	BUF1	I/O buffer IN (31 to 16)	BUF1	I/O buffer OUT (31 to 16)
PCL U-axis (Axis 4)	+18	MSTS	Main status	CMD	Command
	+1a	SSTS	Sub-status	OTP	Not used (reserved)
	+1c	BUF0	I/O buffer IN (15 to 0)	BUF0	I/O buffer OUT (15 to 0)
	+1e	BUF1	I/O buffer IN (31 to 16)	BUF1	I/O buffer OUT (31 to 16)
PCL V-axis (Axis 5)	+20	MSTS	Main status	CMD	Command
	+22	SSTS	Sub-status	OTP	Not used (reserved)
	+24	BUF0	I/O buffer IN (15 to 0)	BUF0	I/O buffer OUT (15 to 0)
	+26	BUF1	I/O buffer IN (31 to 16)	BUF1	I/O buffer OUT (31 to 16)
PCL W-axis (Axis 6)	+28	MSTS	Main status	CMD	Command
	+2a	SSTS	Sub-status	OTP	Not used (reserved)
	+2c	BUF0	I/O buffer IN (15 to 0)	BUF0	I/O buffer OUT (15 to 0)
	+2e	BUF1	I/O buffer IN (31 to 16)	BUF1	I/O buffer OUT (31 to 16)
PCL A-axis (Axis 7)	+30	MSTS	Main status	CMD	Command
	+32	SSTS	Sub-status	OTP	Not used (reserved)
	+34	BUF0	I/O buffer IN (15 to 0)	BUF0	I/O buffer OUT (15 to 0)
	+36	BUF1	I/O buffer IN (31 to 16)	BUF1	I/O buffer OUT (31 to 16)
PCL B-axis (Axis 8)	+38	MSTS	Main status	CMD	Command
	+3a	SSTS	Sub-status	OTP	Not used (reserved)
	+3c	BUF0	I/O buffer IN (15 to 0)	BUF0	I/O buffer OUT (15 to 0)
	+3e	BUF1	I/O buffer IN (31 to 16)	BUF1	I/O buffer OUT (31 to 16)
Option ports	+80	ELPOL	Status of ELS polarity for each axis	ELPOL	ELS polarity setting for each axis
	+82	DLS/PCS	Status of DLS/PCS input selection	DLS/PCS	DLS/PCS input selection
	+84	C4STA	STA output when CMP4 condition is established	C4STA	STA output when CMP4 condition is established
	+86	C5STP	STP output when CMP5 condition is established	C5STP	STP output when CMP5 condition is established
	+8a	J3COUT	Selected signal(3 – 5) J3 output setting status	J3COUT	Selected signal(3 – 5) J3 setting status
	+8c	COTSEL1	X – Z : CMP3 – 5 output selection status	COTSEL1	X – Z : CMP3 – 5 output selection status
	+8e	COTSEL2	V – B : CMP3 – 5 output selection status	COTSEL2	V – B : CMP3 – 5 output selection status
	+90	BINTM	Board interrupt output mask setting status	BINTM	Board interrupt output mask setting
	+92	BINTS	Board interrupt status	BINTR	Not used (reserved)
	+9c	BID	Board ID (setting by DIP SW : 0 -15)	---	Not used
	+c8,ca	BCOD	Board code : 57h, 8ah	---	Not used

Table 3. 2-1 Board Addresses

3.2.2 Option Ports

Option ports are provided with each Board. The port functions are described below.

(1) Sets or reads the ELS polarity setting for each axis (80h : ELPOL).
N. C. condition selected when bit n is set to 0. N. O. condition selected when bit n is set to 1.
N. O. : ELS detected when current to coupler is ON.
N. C. : ELS detected when current to coupler is OFF.

Bit	7	6	5	4	3	2	1	0
Function	BELS	AELS	WELS	VELS	UELS	ZELS	YELS	XELS

(2) Sets or reads the DLS/PCS input selection (82h : DLS/PCS). When the power is turned ON, the bits are set to 0 (DLS).
Selects the DLS signal input as PCS signal input for connector J1 (X to B axes).
DLS input port selected when bit n is set to 0.
PCS input port selected when bit n is set to 1.

Bit	7	6	5	4	3	2	1	0
Function	BPCS	APCS	WPCS	VPCS	UPCS	ZPCS	YPCS	XPCS

(3) Sets or reads the simultaneous start signal (STA) when the Comparator 4 (CMP4) comparison condition is established.
(84h : C4STA) When the power is turned ON, the bits are set to 0 (output disabled).
STA output disabled when bit n is set to 0.
STA output enabled when bit n is se to 1.

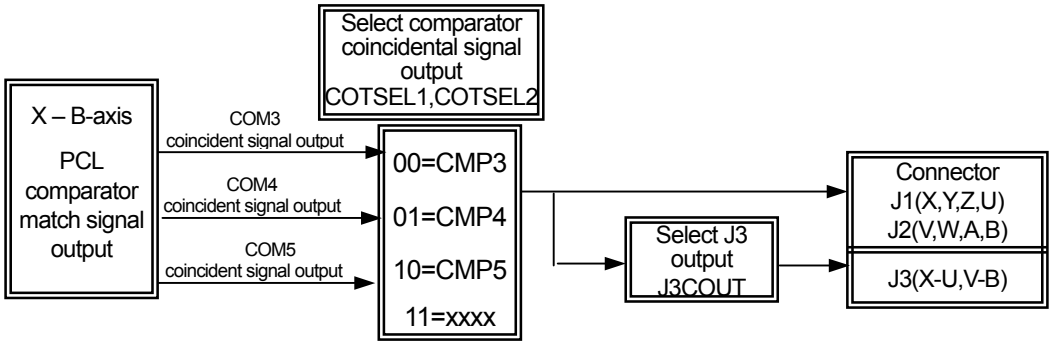
Bit	7	6	5	4	3	2	1	0
Function	B-axis	A-axis	W-axis	V-axis	U-axis	Z-axis	Y-axis	X-axis

(4) Sets or reads the simultaneous stop signal (STP) when the Comparator 5 (CMP5) comparison condition is established.(86h : C5STP) When the power is turned ON, the bits are set to 0 (output disabled).
STP output disabled when bit n is set to 0.
STP output enabled when bit n is se to 1.

Bit	7	6	5	4	3	2	1	0
Function	B-axis	A-axis	W-axis	V-axis	U-axis	Z-axis	Y-axis	X-axis

(5) Sets and reads selection of coincidental output signal with Comparator 3 –5

Bit	7	6	5	4	3	2	1	0
COTSEL1 (8ch)	X-U:Select coincidental output signal with CMP3 - 5							
	U-axis		Z-axis		Y-axis		X-axis	
	00		01		10		11	
	CMP3		CMP4		CMP5		Output enabled	
COTSEL2 (8eh)	V-B: Select coincidental output signal with CMP3 - 5							
	B-axis		A-axis		W-axis		V-axis	
	00		01		10		11	
	CMP3		CMP4		CMP5		Output enabled	
J3COUT (8ah)	Selected signal(3 - 5) J3 output setting							
	B-axis	A-axis	W-axis	V-axis	U-axis	Z-axis	Y-axis	X-axis



(6) Sets or reads the Board interrupt enable (90h : BINTM). When the power is turned ON, bit 0 is set to 0 (interrupts masked).
Sets the interrupt mask from the Board to the PCI bus.
Interrupts masked when bit 0 is set to 0 (interrupt disabled).
Interrupts unmasked when bit 0 is set to 1 (interrupt enabled).

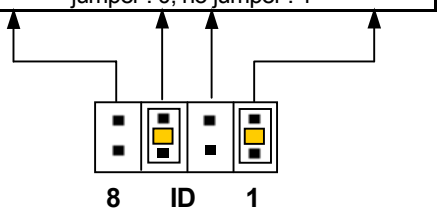
Bit	7	6	5	4	3	2	1	0
Function	0	0	0	0	0	0	0	BINTM

(7) Indicates the Board interrupt status (92h : BINTS). When the power is turned ON, bit 0 is set to 1 (no interrupt).
Indicates the interrupt status from the Board to the PCI bus.
There is an interrupt when bit 0 is 0.
There is no interrupt when bit 0 is 1.

Bit	7	6	5	4	3	2	1	0
Function	0	0	0	0	0	0	0	BINTS

(8) Board ID
Reading board ID by addressing 9ch. The illustrated figure shows resulted value for example.
The factory setting is all jumpered ('0000').
Reads setting status of board ID jumper on board.
There is an jumper when bit n is 0.
There is no jumper when bit n is 1.

Bit	7	6	5	4	3	2	1	0
Function	x	x	x	x	jumper : 0, no jumper : 1			



3.2.3 Board I/O and Device Drivers

The following table shows the correspondence between the device drivers and the I/O to option ports and each axis (PCL6045) on the Board.

Classification	Axis specification or Port	Read (INP)		Write (OUT)	
		Name	Driver function	Name	Driver function
Axis	Argument: Axis 0(X) to7(B)	MSTS	cp530_rMstsW()	CMD	cp530_wCmdW()
		SSTS	cp530_rSstsW()	OTP	--------
		BUF0	cp530_rReg() cp530_rBufDW()	BUF0	cp530_wReg() cp530_wBufDW()
		BUF1		BUF1	
Option port	Argument : Port 0x80 to 0x8e 0x9c, 0xc8, 0xca	ELPOL	cp530_rPortB()	ELPOL	cp530_wPortB()
		DLS/PCS		DLS/PCS	
		CMP4		CMP4	
		CMP5		CMP5	
	0x90	BINTM	(Device driver process)	BINTM	(Device driver
	0x92	BINTS			---------

Table 3. 2-2 Board I/O and Driver Functions

Note 1. There are device drivers provided for each type of OS that can be used to operate the application programs.
Note 2. The device driver's function names are the same for all of the OS versions.
Note 3. Refer to the *CPD Board Series User's Manual: Software Information* for details on the driver functions. The various functions are listed in this manual in *6. Driver Functions*.

3.2.4 PCI configuration registers

The table 3. 2-3 sows the PCI configuration register values for CPD578.

31 24	23 16	15 8	7 0	Address
Device ID 1014h		**Vendor ID 14a9h**		00h
Device status		Device control		04h
Class code			Revision ID (02h)	08h
Base class (06h)	Sub class (80h)	Program interface		
Self test	Header type	Master latency timer	Cash line	0ch
Base address register	0 0 0 0 0 0 0 0 h (Reserved)			10h
	x x x x x x x x h (Reserved for the board)			14h
	0 0 0 0 0 0 0 0 h (Reserved)			18h
	I/O address in the CPD578board			1ch
	0 0 0 0 0 0 0 0 h (Reserved)			20h
	0 0 0 0 0 0 0 0 h (Reserved)			24h
Card bus CIS pointer				28h
Sub system ID 1014h		Sub system vender ID 14a9h		2ch
Reserved				30h | fch

Table 3. 2-3 PCI configuration register

3.3 Board Settings

3.3.1 CPD578 Board

There are two kind of jumper settings on the CPD578 Board: a set of jumper pins for the Board ID setting and three sets of jumper pins for the Encoder circuit format.

The Board ID setting is used to distinguish between Boards when 2 or more Boards are being used in an application program.

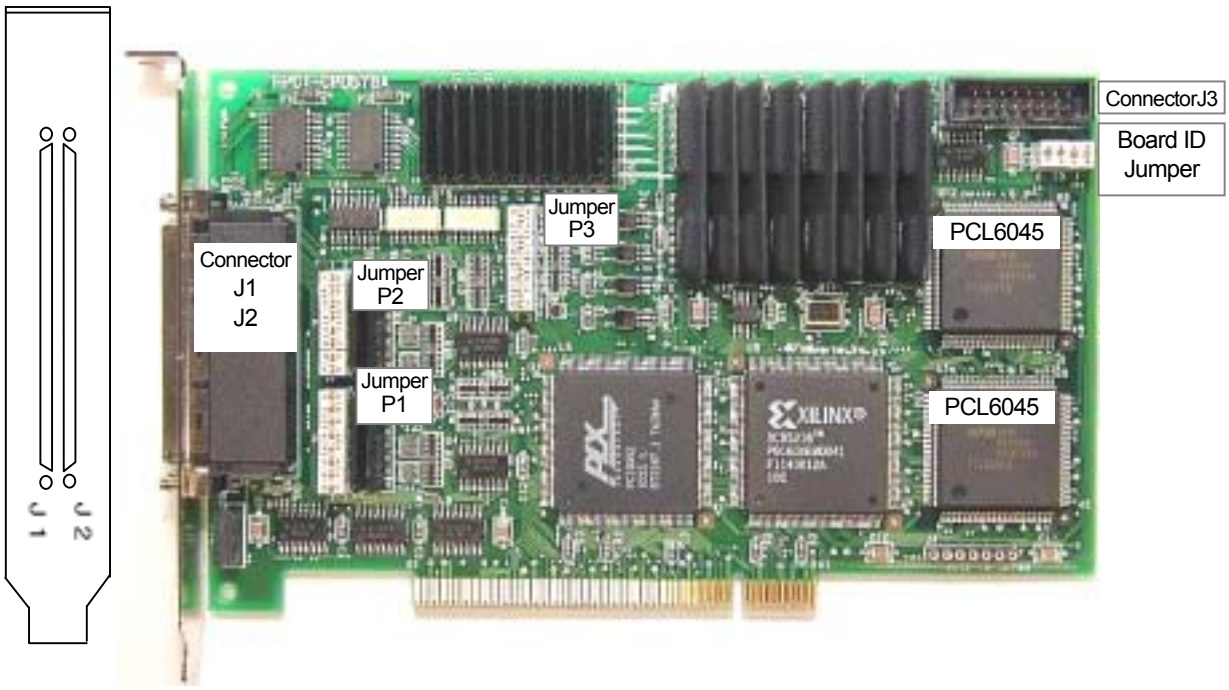


Figure 3. 3-1 Jumper Pin Locations on the CPD578 Board

(1) Board ID Setting Jumper Pins

The Board ID jumper pins can be used to set the CPD Board's Board ID (0 to 15). The following table shows 4 example settings. (The factory setting is 0.)

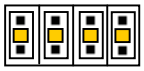
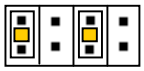
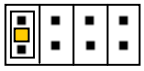
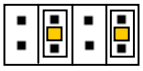
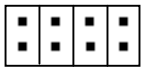
Board ID setting	0	5	7	10	15
Jumper settings	 8 ID 1	 8 ID 1	 8 ID 1	 8 ID 1	 8 ID 1
Binary equivalent	0 0 0 0	0 1 0 1	0 1 1 1	1 0 1 0	1 1 1 1

Table 3. 3-1 Board ID Setting

(2) Encoder Circuit Format (Refer to *3. 4-3 Encoder Input Circuits* for details.)

Encoder A-phase and B-phase are input in differential format. Select the termination setting by Jumper P1 and P2. Select the appropriate input circuit for the encoder output circuit being used (differential or open-collector output). (Refer to *3. 4-3 Encoder Input Circuits* for details.)

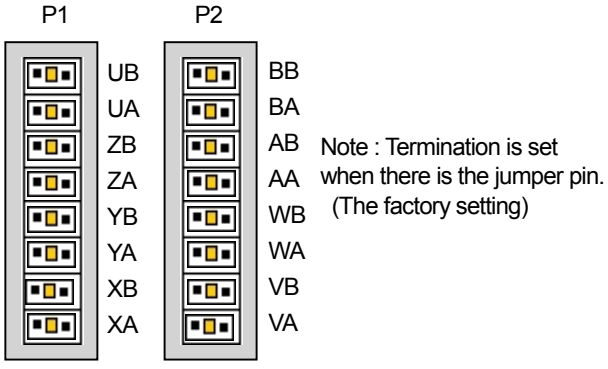


Figure 3. 3-2 Termination setting of encoder A-phase and B-phase

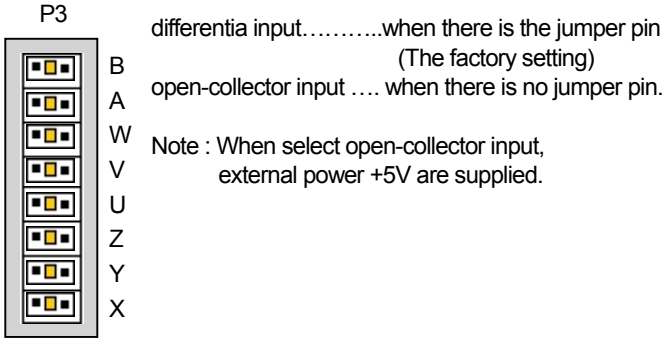


Figure 3. 3-3 Encoder Z-phase input selection

(3) Emergency Stop Function
Can select the following function by cut pattern “E101” on the solder side of the board.

- Function: All axes will stop when the EMG signal is input under operation. None of the axes will operate while the EMG signal is being input.
- Settings: The J1 connector’s XSVALM terminal or J2 connector’s YSVALM terminal is used as the EMG signal input terminal. To connect this terminal to the EMG signal, cut the cut pattern “E101. ” This cut pattern is displayed on the Board. When this pattern is cut, the XSVALM function and YSVALM function will be disabled.
- Logic: The EMG signal is a N. C. (normally closed) condition, so the EMG signal is detected when the photo coupler current is OFF.
- Status: When an EMG signal occurs, MSTS bit 4 (SERR) will be “1” and ERST bit 9 will be “1. ” The input status can be confirmed by verifying that RSTS (extension status register) bit 7 is “1. ”

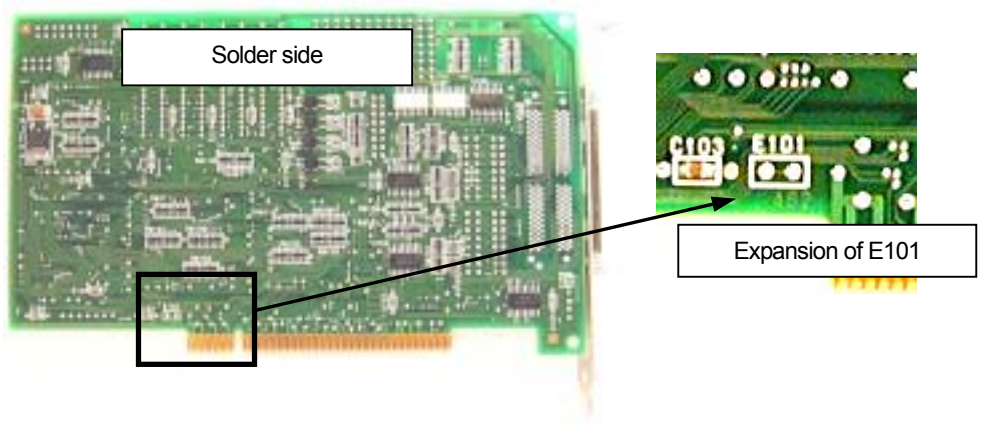
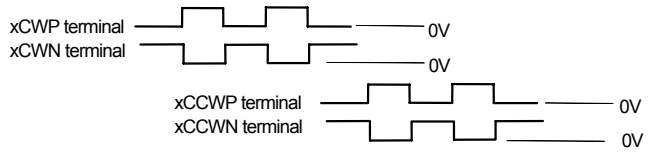
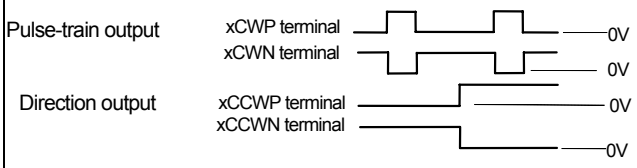
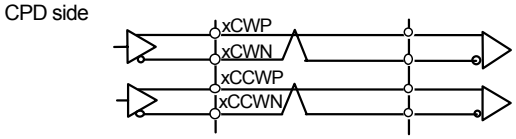
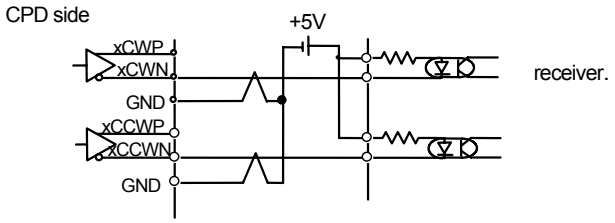
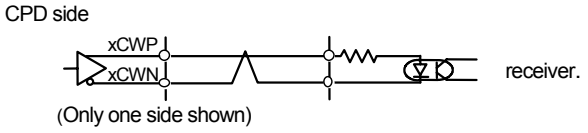
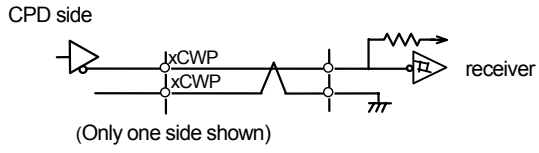


Figure 3. 3-4 “E101” jumper for setting of ‘emergency stop’

3.4 Servo and Machine Interfaces

3.4.1 Command Pulse Output and Driver Connections

Table 3. 4-1 shows the output circuit conditions.

No.	Item		Contents
1	Electrical conditions	Output pulse driver pulse width	Differential driver (26LS31) Command pulse frequency 50% duty width or a pulse width of 200 microsecond when the frequency is less than 2. 4 Kpps
2	Signal format	Individual pulse output method (Set in Environment Setting Register 1.)	
		Direction and pulse train method (Set the direction output logic in Environment Setting Register 1.)	
3	Connection to motor driver	Connection to motor driver by differential input	
		Connection to motor driver by coupler input	
		For the motor drivers ensuring differential connection to the coupler input	
		Connection to motor driver by TTL input	

Note: Caution is required for speeds and cable lengths when the motor driver does not use a differential input. As guidelines, use 500 Kpps with a cable length of 3 m for coupler connections and 250 Kpps with a cable length of 1 m for TTL connections. Always check motor driver reception circuit specifications before application.

Table 3. 4-1 Command Pulse Output Circuits

3.4.2 Axis Sensor and Servo Interface Input Circuits

Table 3. 4-2 shows the input circuits.

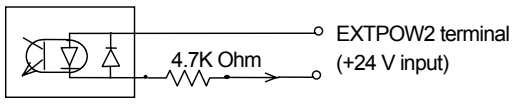
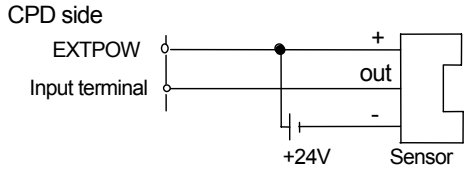
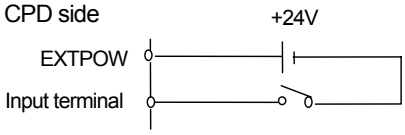
No.	Item		Contents
1	Circuit format 1	Shared by ±xELS, xDLS, and xOLS	 EXTPOW1: standard 24-V input terminal ■ Following Resistance Module needed to mount when using 12V
2	Circuit format 2	Shared by XINPOS, xSVALM	 Input terminal: 5 mA ≤ IS ≤ 10 mA EXTPOW2: standard 24-V input terminal
3	Polarity settings	Polarity settings for ±xELS, xDLS, xOLS, xINPOS, and xSVALM	N. O. (Normal Open) condition: The signal is detected when current to the photo coupler is ON N. C. (Normal Close) condition: The signal is detected when current to the photo coupler is OFF. For ELS, set the polarity with the option port. (See *3. 2. 2 Option Ports* for details.) For DLS, OLS, INPOS, and SVALM, set the polarity with the RENV1. (See *CPD Board Series User's Manual: Common Information* for details.)
4	External connections	Photoelectric sensor input	
		Limit switch input	

Table 3. 4-2 Axis Sensor and Servo Interface Input Circuits

3.4.3 Encoder Input Circuits

Table 3. 4-3 shows the input circuit conditions.

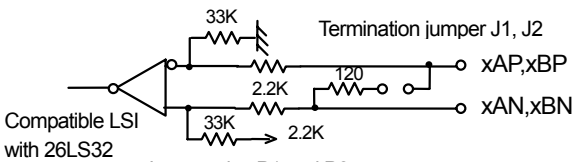
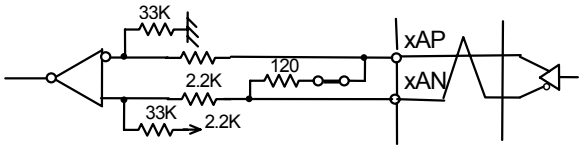
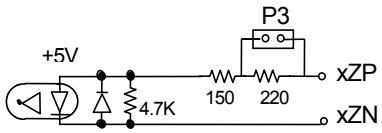
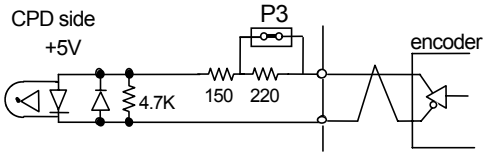
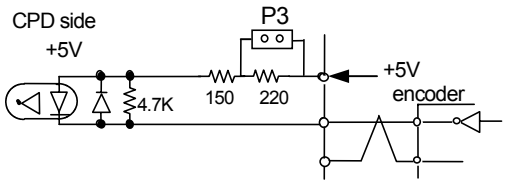
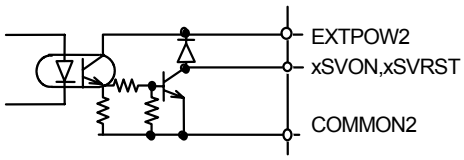
No.	Item		Contents
1	Encoder Phase A or Phase B	Encoder input circuit format	 Jumper pins P1 and P2 Encoder circuit format setting Note: As input circuit of A, B-phase, Connect plus terminal to (-) side of 26LS32 for reversing internal logic. When used differential driver, connect (+) to P, (-) to N as above.
		Advancement of phase settings	Set by means of software.
		Differential connection	
2	Encoder Phase-Z	Encoder input circuit format	
		Differential connection	
		Open-collector connection (Supply +5V from an external power source and leave the jumper off)	

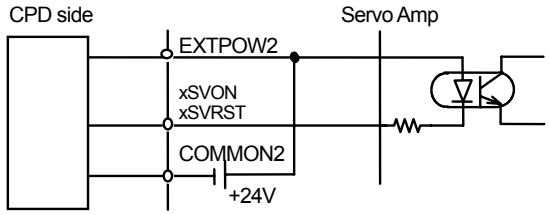
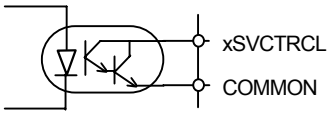
Table 3. 4-3 Encoder Input Circuit

3.4.4 Servo Interface Output Circuits

Table 3. 4-4 shows the output circuit conditions.

<table>
<tr><th>No.</th><th colspan="2">Item</th><th>Contents</th></tr>
<tr><td rowspan="3">1</td><td rowspan="3">xSVON
(servo ON)

xSVRST
(servo reset)</td><td>Driver circuit format</td><td> 

Rating load voltage: DC12 - DC24V
Allowable load current: Less than 80mA/terminal (DC12V – DC24V)</td></tr>
<tr><td>Output logic level
(The polarity cannot be changed.)</td><td>The xSVON, xSVRST is ON when the port output is “1”.</td></tr>
<tr><td>External connections</td><td> 
</td></tr>
<tr><td rowspan="3">2</td><td rowspan="3">xSVCTRCL
(error counter clear)</td><td>Driver circuit format</td><td> 

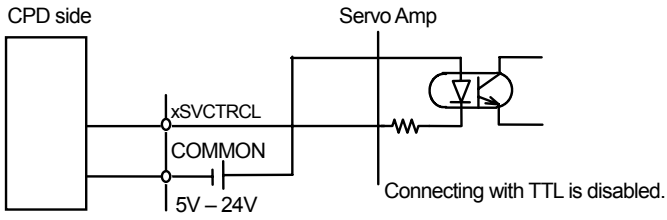
Rating load voltage: DC5 - DC24V
Allowable load current: Less than 16mA/terminal</td></tr>
<tr><td>Output logic level
(The polarity cannot be changed.)</td><td>The xSVCTRCL is ON when the port output is “1”.</td></tr>
<tr><td>External connections</td><td> 
</td></tr>
</table>

Table 3. 4-4 Servo Interface Output Circuits

3.4.5 Simultaneous Start/Stop and Match Signal Output

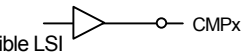
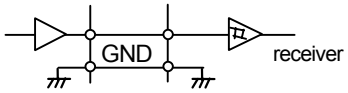
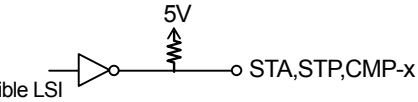
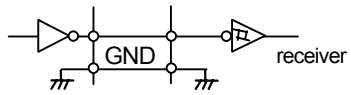
No.	Item		Contents
1	J1 / J2 Match signal output J1: CMP-X – CMP-U J2: CMP-V – CMP-B	Circuit format	 Compatible LSI with 74LS07 Rating load voltage: TypDC5V Rating load current: Less than 30mA
		Output logic levels	'HIGH' when port output is active. (Active high)
		External connections	
2	J3 Match signal output CMP-X – CMP-B Simultaneous Start STA Simultaneous Stop STP	Circuit format	 Compatible LSI With 74HC04 Rating load voltage: DC5V Rating load current: Less than 6mA
		Output logic levels	'LOW' when port output is active. (Active low)
		External connections	

Table 3. 4-5 J1 – J3:Match signal and J3:Simultance Start/Stop Output Circuits

3.5 Connector PIN out

3.5.1 CPD578 connector PIN out

Connector J1 and J2 are placed as following figure

J1 is allotted to signal of X – U-axes,and J2 is allotted to signal of V –B-axes.

PIN No.	Signal Name	PIN No.	Signal Name
1	+5V power out	51	GND
2	+5V power out	52	GND
3	XCWP (CW pulse out +)	53	ZCWP (CW pulse out +)
4	XCWN (CW pulse out -)	54	ZCWN (CW pulse out -)
5	XCCWP (CCW pulse out +)	55	ZCCWP (CCW pulse out +)
6	XCCWN (CCW pulse out -)	56	ZCCWN (CCW pulse out -)
7	YCWP (CW pulse out +)	57	UCWP (CW pulse out +)
8	YCWN (CW pulse out -)	58	UCWN (CW pulse out -)
9	YCCWP (CCW pulse out +)	59	UCCWP (CCW pulse out +)
10	YCCWN (CCW pulse out -)	60	UCCWN (CCW pulse out -)
11	XAP(Encoder A-phase input +)	61	ZAP(Encoder A-phase input +)
12	XAN(Encoder A-phase input -)	62	ZAN(Encoder A-phase input -)
13	XBP(Encoder B-phase input +)	63	ZBP(Encoder B-phase input +)
14	XBN(Encoder B-phase input -)	64	ZBN(Encoder B-phase input -)
15	XZP(Encoder Z-phase input +)	65	ZZP(Encoder Z-phase input +)
16	XZN(Encoder Z-phase input -)	66	ZZN(Encoder Z-phase input -)
17	YAP(Encoder A-phase input +)	67	UAP(Encoder A-phase input +)
18	YAN(Encoder A-phase input -)	68	UAN(Encoder A-phase input -)
19	YBP(Encoder B-phase input +)	69	UBP(Encoder B-phase input +)
20	YBN(Encoder B-phase input -)	70	UBN(Encoder B-phase input -)
21	YZP(Encoder Z-phase input +)	71	UZP(Encoder Z-phase input +)
22	YZN(Encoder Z-phase input -)	72	UZN(Encoder Z-phase input -)
23	GND	73	COMMON (Common GND for xSVCTRCL)
24	GND	74	COMMON (Common GND for xSVCTRCL)
25	XSVCTRCL(Error counter clear output)	75	ZSVCTRCL(Error counter clear output)
26	YSVCTRCL(Error counter clear output)	76	USVCTRCL(Error counter clear output)
27	XSVALM (Sarvo alarm input)	77	ZSVALM (Sarvo alarm input)
28	XINPOS (Positioning completion input)	78	ZINPOS (Positioning completion input)
29	XSVON (Sarvo ON output)	79	ZSVON (Sarvo ON output)
30	XSVRST (Sarvo reset output)	80	ZSVRST (Sarvo reset output)
31	YSVALM (Sarvo alarm input)	81	USVALM (Sarvo alarm input)
32	YINPOS (Positioning completion input)	82	UINPOS (Positioning completion input)
33	YSVON (Sarvo ON output)	83	USVON (Sarvo ON output)
34	YSVRST (Sarvo reset output)	84	USVRST (Sarvo reset output)
35	EXTPOW2 (+24V power input)	85	COMMON2 (Common GND for +24V)
36	EXTPOW2 (+24V power input)	86	COMMON2 (Common GND for +24V)
37	＋XELS (Input for positive limit sensor)	87	＋ZELS (Input for positive limit sensor)
38	－XELS (Input for negative limit sensor)	88	－ZELS (Input for negative limit sensor)
39	XDLS (Input for decelerate limit sensor)	89	ZDLS (Input for decelerate limit sensor)
40	XOLS (Input for origin sensor)	90	ZOLS (Input for origin sensor)
41	＋YELS (Input for positive limit sensor)	91	＋UELS (Input for positive limit sensor)
42	－YELS (Input for negative limit sensor)	92	－UELS (Input for negative limit sensor)
43	YDLS (Input for decelerate limit sensor)	93	UDLS (Input for decelerate limit sensor)
44	YOLS (Input for origin sensor)	94	UOLS (Input for origin sensor)
45	EXTPOW1 (+24V power input)	95	EXTPOW1 (+24V power input)
46	EXTPOW1 (+24V power input)	96	EXTPOW1 (+24V power input)
47	+5V power out	97	GND
48	+5V power out	98	GND
49	CMPX output (TTL) (TTL OC output)	99	CMPZ output (TTL OC output)
50	CMPY output (TTL OC output)	100	CMPU output (TTL OC output)

Table 3. 5-1 CPD578 J1 CONNECTOR PIN OUT (Xaxis - Uaxis)

・ Connector type

Board side:HDRA-E100W
1LFDT1EC-SL
(HONDA TSUSHIN)

Cable side:HDRA-E100MA1(connector)
HDRA-E100LPWC(case)

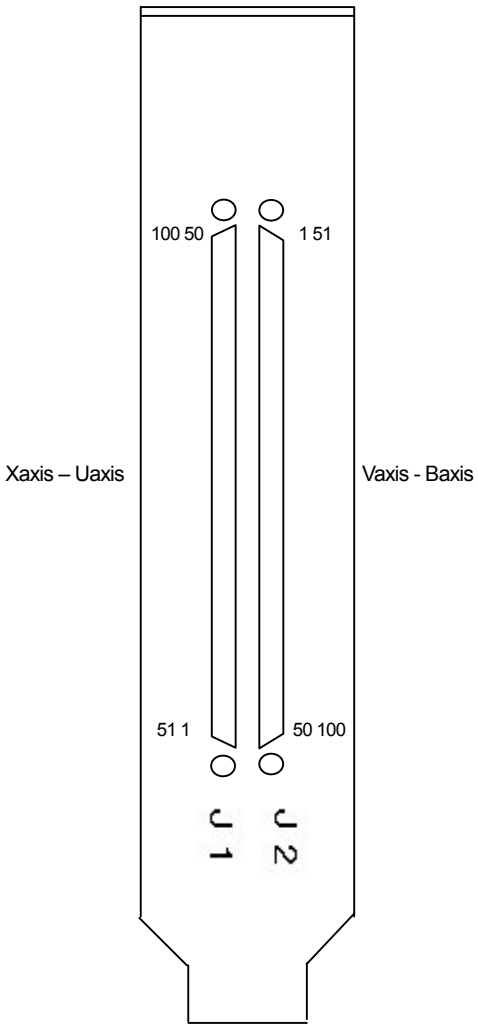


Figure 3. 5-1 Front panel

PIN No.	Signal Name	PIN No.	Signal Name
1	+5V power out	51	GND
2	+5V power out	52	GND
3	VCWP (CW pulse out +)	53	ACWP (CW pulse out +)
4	VCWN (CW pulse out -)	54	ACWN (CW pulse out -)
5	VCCWP (CCW pulse out +)	55	ACCWP (CCW pulse out +)
6	VCCWN (CCW pulse out -)	56	ACCWN (CCW pulse out -)
7	WCWP (CW pulse out +)	57	BCWP (CW pulse out +)
8	WCWN (CW pulse out -)	58	BCWN (CW pulse out -)
9	WCCWP (CCW pulse out +)	59	BCCWP (CCW pulse out +)
10	WCCWN (CCW pulse out -)	60	BCCWN (CCW pulse out -)
11	VAP(Encoder A-phase input +)	61	AAP(Encoder A-phase input +)
12	VAN(Encoder A-phase input -)	62	AAN(Encoder A-phase input -)
13	VBP(Encoder B-phase input +)	63	ABP(Encoder B-phase input +)
14	VBN(Encoder B-phase input -)	64	ABN(Encoder B-phase input -)
15	VZP(Encoder Z-phase input +)	65	AZP(Encoder Z-phase input +)
16	VZN(Encoder Z-phase input -)	66	AZN(Encoder Z-phase input -)
17	WAP(Encoder A-phase input +)	67	BAP(Encoder A-phase input +)
18	WAN(Encoder A-phase input -)	68	BAN(Encoder A-phase input -)
19	WBP(Encoder B-phase input +)	69	BBP(Encoder B-phase input +)
20	WBN(Encoder B-phase input -)	70	BBN(Encoder B-phase input -)
21	WZP(Encoder Z-phase input +)	71	BZP(Encoder Z-phase input +)
22	WZN(Encoder Z-phase input -)	72	BZN(Encoder Z-phase input -)
23	GND	73	COMMON (Common GND for xSVCTRCL)
24	GND	74	COMMON (Common GND for xSVCTRCL)
25	VSVCTRCL(Error counter clear output)	75	ASVCTRCL(Error counter clear output)
26	WSVCTRCL(Error counter clear output)	76	BSVCTRCL(Error counter clear output)
27	VSVALM (Sarvo alarm input)	77	ASVALM (Sarvo alarm input)
28	VINPOS (Positioning completion input)	78	AINPOS (Positioning completion input)
29	VSVON (Sarvo ON output)	79	ASVON (Sarvo ON output)
30	VSVRST(Sarvo reset output)	80	ASVRST(Sarvo reset output)
31	WSVALM (Sarvo alarm input)	81	BSVALM (Sarvo alarm input)
32	WINPOS (Positioning completion input)	82	BINPOS (Positioning completion input)
33	WSVON (Sarvo ON output)	83	BSVON (Sarvo ON output)
34	WSVRST(Sarvo reset output)	84	BSVRST(Sarvo reset output)
35	EXTPOW2 (+24V power input)	85	COMMON2 (Common GND for +24V)
36	EXTPOW2 (+24V power input)	86	COMMON2 (Common GND for +24V)
37	＋VELS (Input for positive limit sensor)	87	＋AELS (Input for positive limit sensor)
38	－VELS (Input for negative limit sensor)	88	－AELS (Input for negative limit sensor)
39	VDLS (Input for decelerate limit sensor)	89	ADLS (Input for decelerate limit sensor)
40	VOLS (Input for origin sensor)	90	AOLS (Input for origin sensor)
41	＋WELS (Input for positive limit sensor)	91	＋BELS (Input for positive limit sensor)
42	－WELS (Input for negative limit sensor)	92	－BELS (Input for negative limit sensor)
43	WDLS (Input for decelerate limit sensor)	93	BDLS (Input for decelerate limit sensor)
44	WOLS (Input for origin sensor)	94	BOLS (Input for origin sensor)
45	EXTPOW1 (+24V power input)	95	EXTPOW1 (+24V power input)
46	EXTPOW1 (+24V power input)	96	EXTPOW1 (+24V power input)
47	+5V power out	97	GND
48	+5V power out	98	GND
49	CMPV output (TTL OC output)	99	CMPA output (TTL OC output)
50	CMPW output (TTL OC output)	100	CMPB output (TTL OC output)

Table 3. 5-2 CPD578 J2 CONNECTOR PIN OUT (Vaxis - Baxis)

PIN No.	Signal Name	PIN No.	Signal Name
1	GND	2	GND
3	STA(Simultaneous Start)	4	STP(Simultaneous Stop)
5	GND	6	GND
7	CMP-X(X-axis comparator match signal)	8	CMP-Y(Y-axis comparator match signal)
9	CMP-Z(X-axis comparator match signal)	10	CMP-U(U-axis comparator match signal)
11	GND	12	GND
13	CMP-V(V-axis comparator match signal)	14	CMP-W(W-axis comparator match signal)
15	CMP-A(A-axis comparator match signal)	16	CMP-B(B-axis comparator match signal)

Table 3. 5-3 CPD578 J3 CONNECTOR PIN OUT

3.6 HPCI-CPD578 Specification

	Item		Specification	Notes
Basic Specifications	Axis control	(1)Positioning system	1 to 8 axes	Control LSI: PCL6045 series
		(2)Linear interpolation system	2 to 4 axes simultaneously x2	
		(3)Circular interpolation system	2 axes simultaneously x2	
		(4)Pulsar feed system	1 to 8 axes	
	Command system	Command pulses	Pulse-train output by differential driver	
		Position command range	-134,217,728 to 134,217,727 pulses	
		Command coordinate	Relative coordinate command	
		Reference range at continuous feed	Unlimited (but not applicable to the position range of the counter	Available of the cyclic counting for the position every 360deg.
		Position override	Possible only in positioning system	
	Speed Control	Speed range	0. 1pps to 6. 5Mpps (Multiplication 0. 1 to 100)	
		Constant control of linear speed	2-axes circular, linear interpolation : :square root method 3-axes linear interpolation : cubic root method 4-axes linear interpolation : cubic root method	
		Speed override	(1)Constant feed(without acc/dec.): available in all cases (2)Limited to the positioning, linear interpolation and continuous feed. allowing it during acc/dec duration.	
	Acceleration/ deceleration control (Acc/Dec control)	Automatic Acc/Dec system	(1)Positioning and linear interpolation: Linear acc/dec system, S-curve acc/dec system and the partial S-curve acc/dec system are available. ・Triangular drive peak prevention function available ・Asymmetrically sloped acc/dec possible. (2)For circular interpolation: Automatic acc/dec possible except S-curve	
Function Specifications	Origin-returning control	Origin-returning method	13 returning methods for the sensor's origin,Z-phase origin,ELS shared origin (Each returning is done in a single process by the single command)	
		Origin search	Available	
		Getting out of origin area	Available	
	UP/DOWN counters		Command position counter (Command pulse count: length 28-bit)	
			Machine position counter (Encoder pulse count: length 28-bit)	
			General-purpose counter (Pulse count: length 28-bit)	
			Error counter for runaway detection (Counter: length 16-bit) (Difference between the values of the encoder and the command.)	
	Comparators (CMP)		CMP1 and CMP2 : Soft-limit use comparing positive and negative limits	
			CMP3,4 and 5: General-purpose(used for changing speed or starting another axis on comparison of the coordinate)	
	Inputs for encoder or pulsar		Each axis has a common input terminal for the encoder/pulsar input signal. (Encoder output is differential or open-collector output.)	
	Backlash compensation		Compensated every turn of the direction except the circular interpolation system.	
	Positioning-control-start function (PCS) Idling pulse function		Starts positioning by PCS signal during continuous feed. (Only for positioning system)	
			Function of improving stepping motor acceleration characteristics.	
	Vibration suppress at a stop		Function of vibration suppression when the stepping motor stops.	
	Machine interface		+/-ELS,OLS,DLS, and encoder A,B, and Z-phase per axis. (All with photo coupler isolation)	
	Servo interface		Command-pulse output: differential output	
			Input: Servo alarm and in-position per axis (all with photo coupler insulation)	
			Output: Servo reset, servo ON, and servo error counter clear per axis. (all with photo coupler insulation)	
Conditions	Ambient temperature		0 to 50 centigrade (no condensation)	
	Current consumption		+5V 2A	
	Dimensions		175mm x 107mm (Short size)	

Table 3. 6-1 HPCI-CPD578 Specifications

4. Functions

4.1 Combining Operations Between Axes

One set of the 4-axis PCL6045 LSI has three kinds of the axis operation functions: independent axis operation, linear interpolation of a group of 2 to 4 axes, and circular interpolation of 2 axes.
The “independent axis operations” include the following functions for each axis: Positioning (PTP), continuous feed (terminate with a stop command), origin returning, pulsar feed (handle feed), and timer operation.
In the descriptions about (Axis Operation Combinations), the followings are the meaning of “simultaneous operations”

(1) Operating multiple independent axes at the same time.
    Example: performing X-axis operation, continuing with Y-axis operation…
(2) Operating two pairs of interpolating axes at the same time.
    Example: performing XY circular interpolation, and continuing with ZU linear interpolation….
(3) Operating independent axis and interpolating axes at the same time.
    Example: performing XYZ-axis linear interpolation, and independent U-axis operation …

4.1.1 CPD578 Axis Operation Combinations

There are two 4-axis LSI on this board (X – U-axis and V – B-axis). Each LSI is detached.
Linear interpolation of five to eight axes is possible by using simultaneous start command exceptionally. It can be used under constant speed operation without acceleration/deceleration.

No.	Axis operation combination	X – U-axis	V – B-axis	Use simultaneous start command
1	All axes independent	Simultaneous operation is possible.	Regardless of X -- U-axes operation, V -- B-axes operation No.1 --4combination is possible.	
2	All axes in linear interpolation	Two – four axis linear interpolation is possible.		5 – 8 axes liner–interpolation is possible. (on condition of constant speed operation)
3	Liner–interpolation axes & independent axes	Simultaneous operation is possible.		
4	Circle–interpolation axes & the rest axes	Simultaneous operation of circle–interpolation axes & the rest axes (Liner–interpolation axes or independent axes) is possible.		

Table 4. 1-1 CPD578 Axis Operation Combinations

4.2 Interrupt Mechanisms

MSTS bit 4 (SERR): When an error bit has been set in the error status register (REST), the SERR bit will be set and this is one source of an interrupt output for the affected axis.
MSTS bit 5 (SINT): When a bit has been set in the event status register (RIST) and the bit is not masked, the SINT bit will be set and this is the other source of an interrupt output for the affected axis.

These two interrupt sources from each axis' MSTS register are logically ORed and the result is output as an interrupt from the PCL6045. The interrupt outputs from a CPD578 Board to the PCI bus must pass through the Board's interrupt mask (option port BINTM bit) and then are transmitted to the CPU.
The processing of interrupt signals to the CPU are supported by the DOS device driver, but it is also possible to read the source of the interrupt with MSTS polling and mask Board interrupts.

■ Each axis' interrupts can be enabled or disabled with bit 29 of the axis' environment register 1 (RENV1).

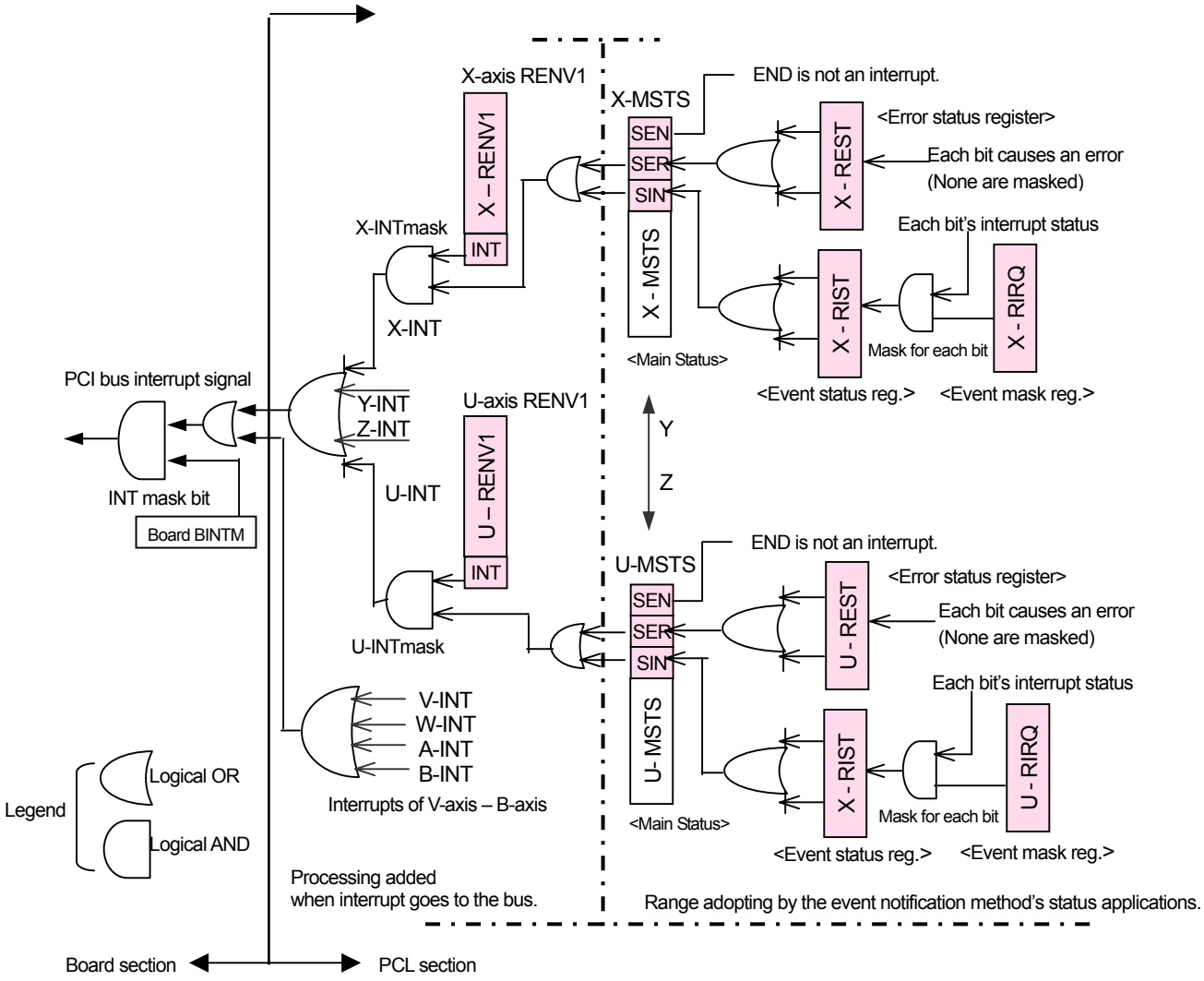


Figure 4. 2-1 Route of Interrupts in the CPD578 Board

5. Software Startup Guide

5.1 Overview

This section explains the following operations.

(1) Windows Version

1. Installing and uninstalling drivers
2. Accessing the Board and the Board ID
3. Using the sample programs
4. Executing and navigating the “Let’s try to move” (an executable file)

(2) DOS Version (MS-DOS or PC DOS)

1. Installing and uninstalling drivers
2. Accessing the Board and the Board ID
3. In the software-related manuals, the CPD578 Boards are known as CPD Boards.

5.2 Software Configuration

(1) Device Drivers

1. There are 3 different device driver files for Windows, depending on the version of Windows being used.
   For Windows XP (abbreviated as WinXP) hc530wxp. sys
   For Windows 2000 (abbreviated as Win2K) hc530w2k. sys
   For Windows NT (abbreviated as WinNT) hicpd530. sys
   For Windows 98 (abbreviated as Win98) hicpd530. vxd
2. There is just one device driver file for DOS.
   DOS (MS-DOS or PC DOS) hicpd530. drv

(2) Device Driver Functions

The various functions included in the device driver interface library are referred to as “driver functions. ”

1. Driver Functions for Windows
   hicpd530. dll (The same file is used for all of the Windows operating systems.)

2. Driver Functions for DOS (Use the same memory model used in the application being created.)
   sicpd530. lib (small model) (Code: less than 64 KB; Data: less than 64 KB)
   cicpd530. lib (compact model) (Code: less than 64 KB; Data: 64 KB or more)
   micpd530. lib (medium model) (Code: 64 KB or more; Data: less than 64 KB)
   licpd530. lib (large model) (Code: 64 KB or more; Data: 64 KB or more)

(3) Library Functions Level 1

A group of “library functions that perform special processing using driver functions” for application programs has been provided in a source program. The contents of these library functions can be changed freely.
These functions are known as “library functions” in contrast to the driver functions.

1. Library Functions for Windows
   cp530l1a. c (cp530l1a. h) For Microsoft Visual C++ (Version 5. 0 or higher)
   cp530l1a. bas For Microsoft Visual Basic (Version 5. 0 / 6. 0)
2. Library Functions for DOS
   cp530l1a. c (cp530l1a. h) For MS-C (6. 0), MS-C++ (7. 0), TC++ (4. 0J), etc.

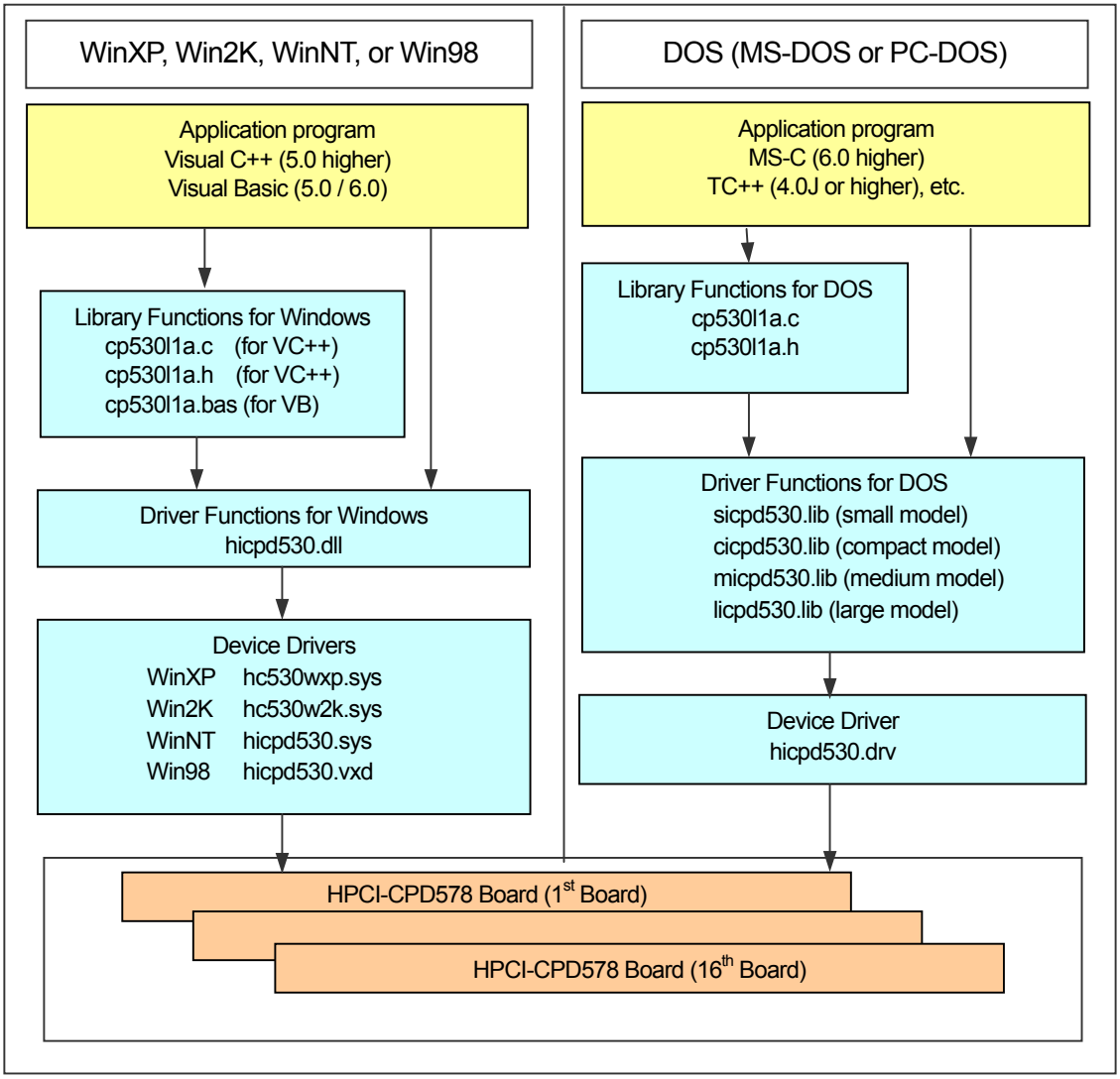


Figure 5. 2-1 Software Configurations

5.3 Installing and Uninstalling the Device Drivers

5.3.1 Installing/Uninstalling Device Drivers in a Windows Systems

(1) Windows XP Installation

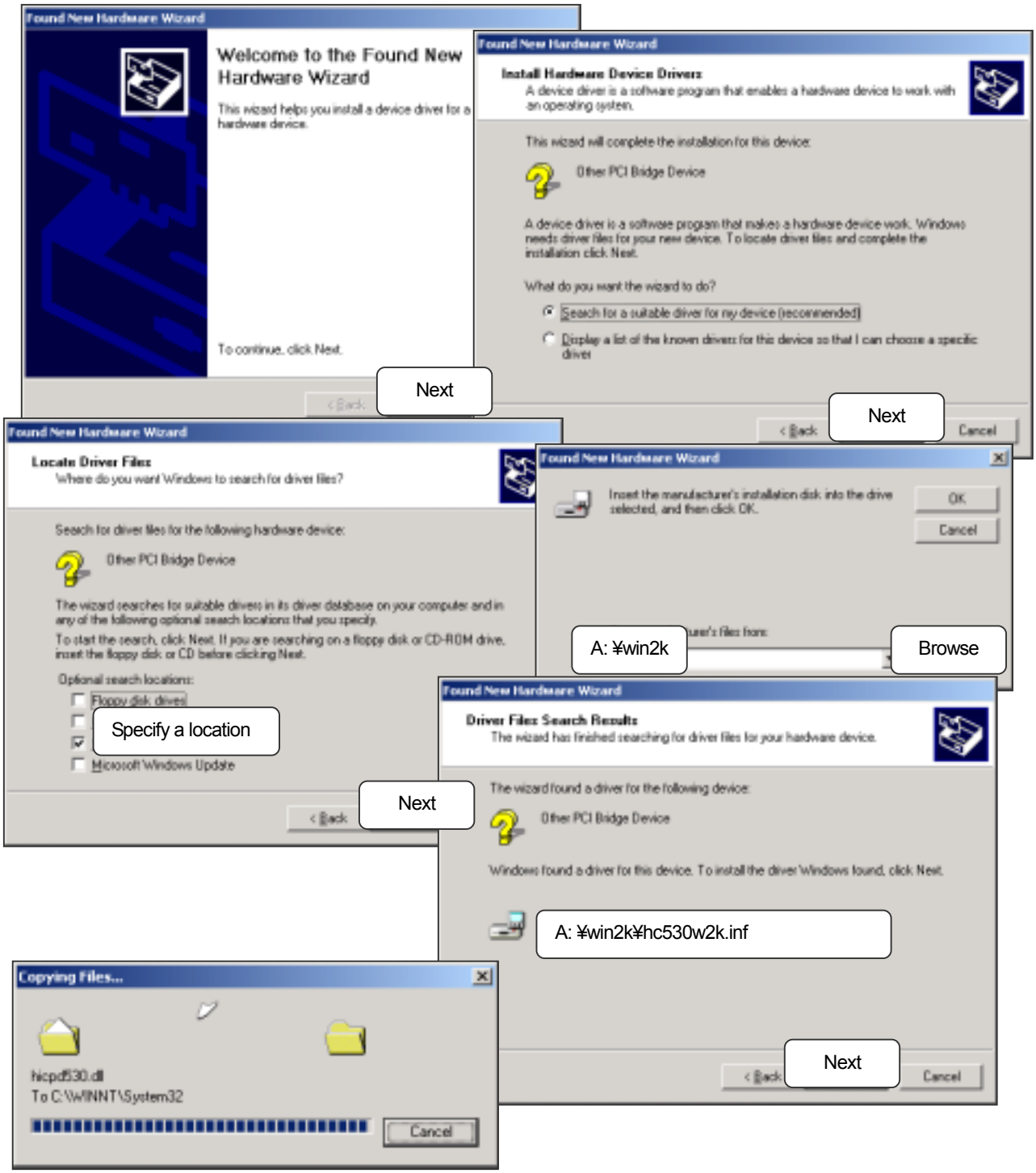
1. Verify that the computer's power supply is OFF and install the CPD Board in an available PCI slot in the computer. Turn ON the computer's power supply and start Windows.
2. When WinXP starts up, the CPD Board will be detected by the system and a window will be displayed automatically to guide you in the installation of the required device driver. A dialog box will be displayed requesting the installation source directory. Insert the provided floppy disk into the computer's floppy disk drive.
3. Check "Install the software automatically (Recommended)".
4. Select "Hivertec HPCI-CPD530 (WinXP)".
5. The warning "has not passed Windows Logo testing to verify its compatibility with Windows XP" is displayed, but continue anyway. We have checked that this software operates on WindowsXP.
Complete the installation by following the directions in the system dialog boxes.

Found New Hardware Wizard
Please select the best match for your hardware from the list below.
Hivertec HPCI-CPD530(WinXP)
This driver is not digitally signed!
Next
Welcome to the Found New Hardware Wizard
Next
Cancel
Completing the Found New Hardware Wizard
Select Continue Anyway.
Hardware Installation
The software you are installing for this hardware:
Hivertec HPCI-CPD530(WinXP)
has not passed Windows Logo testing to verify its compatibility with Windows XP.
Continuing your installation of this software may impair or destabilize the correct operation of your system either immediately or in the future. Microsoft strongly recommends that you stop this installation now and contact the hardware vendor for software that has passed Windows Logo testing.
Continue Anyway
Finish
System Settings Change
Windows has finished installing new devices. The software that supports your device requires that you restart your computer. You must restart your computer before the new settings will take effect.
Do you want to restart your computer now?
Yes

Figure 5. 3-1 WinXP Installations

(2) Windows 2000 Installation

1. Verify that the computer's power supply is OFF and install the CPD Board in an available PCI slot in the computer. Turn ON the computer's power supply and start Windows.
2. When Win2K starts up, the CPD Board will be detected by the system and a window will be displayed automatically to guide you in the installation of the required device driver.
3. A dialog box will be displayed requesting the installation source directory. Insert the provided floppy disk into the computer's floppy disk drive.
4. Check "Specify a location".
5. Specify the directory "A:¥WIN2K". (In this case, the floppy drive is drive A.)
   Complete the installation by following the directions in the system dialog boxes.


Found New Hardware Wizard
Welcome to the Found New Hardware Wizard
This wizard helps you install a device driver for a hardware device.
To continue, click Next.
Next
Found New Hardware Wizard
Install Hardware Device Drivers
A device driver is a software program that enables a hardware device to work with an operating system.
This wizard will complete the installation for this device:
Other PCI Bridge Device
A device driver is a software program that makes a hardware device work. Windows needs driver files for your new device. To locate driver files and complete the installation click Next.
What do you want the wizard to do?
Search for a suitable driver for my device (recommended)
Display a list of the known drivers for this device so that I can choose a specific driver
Next
Cancel
Found New Hardware Wizard
Locate Driver Files
Where do you want Windows to search for driver files?
Search for driver files for the following hardware device:
Other PCI Bridge Device
The wizard searches for suitable drivers in its driver database on your computer and in any of the following optional search locations that you specify.
To start the search, click Next. If you are searching on a floppy disk or CD-ROM drive, insert the floppy disk or CD before clicking Next.
Optional search locations:
Floppy disk drives
Microsoft Windows Update
Specify a location
Next
Found New Hardware Wizard
Insert the manufacturer's installation disk into the drive selected, and then click OK.
OK
Cancel
A: ¥win2k
Browse
Found New Hardware Wizard
Driver Files Search Results
The wizard has finished searching for driver files for your hardware device.
The wizard found a driver for the following device:
Other PCI Bridge Device
Windows found a driver for this device. To install the driver Windows found, click Next.
A: ¥win2k¥hc530w2k.inf
Next
Cancel
Copying Files...
hicpd530.dll
To C:\WINNT\System32
Cancel

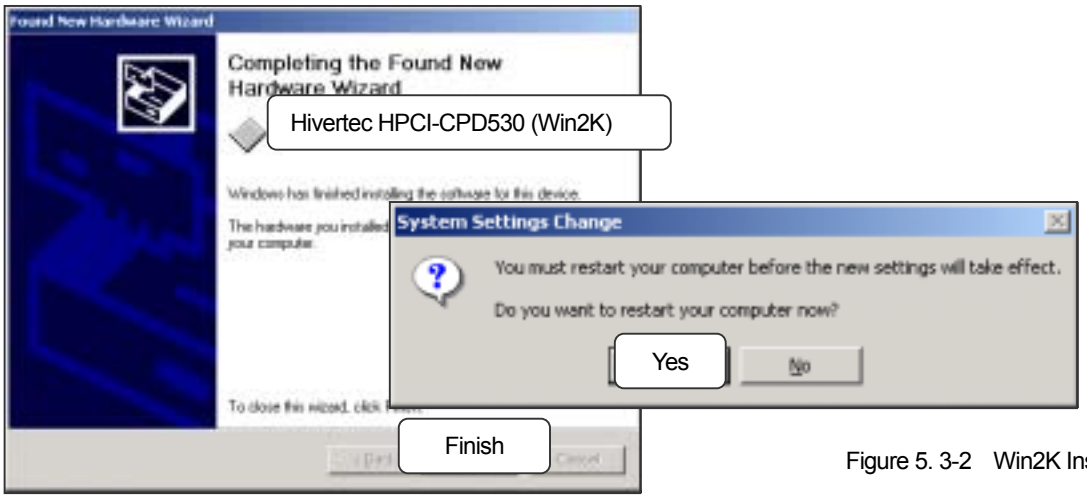

Found New Hardware Wizard
Completing the Found New Hardware Wizard
Hivertec HPCI-CPD530 (Win2K)
System Settings Change
You must restart your computer before the new settings will take effect.
Do you want to restart your computer now?
Yes
No
Finish

Figure 5. 3-2 Win2K Installations

(3) Windows NT 4. 0 Installation

[Device Driver Installation]

1. Insert the provided floppy disk into the computer's floppy disk drive. Start "Windows NT Explorer" and select the file "A:¥WinNT¥c530inst. inf". (In this case, the floppy drive is drive A.)
2. Next click the right mouse button. Select NTLP:***Install*** from the popup menu that is displayed. The device driver installation will begin when NTLP:***Install*** is selected. Complete the installation by following the directions in the system dialog boxes. (It is also possible to start the device driver installation by executing the A:¥WinNT¥c530inst. bat batch file from the command prompt.)

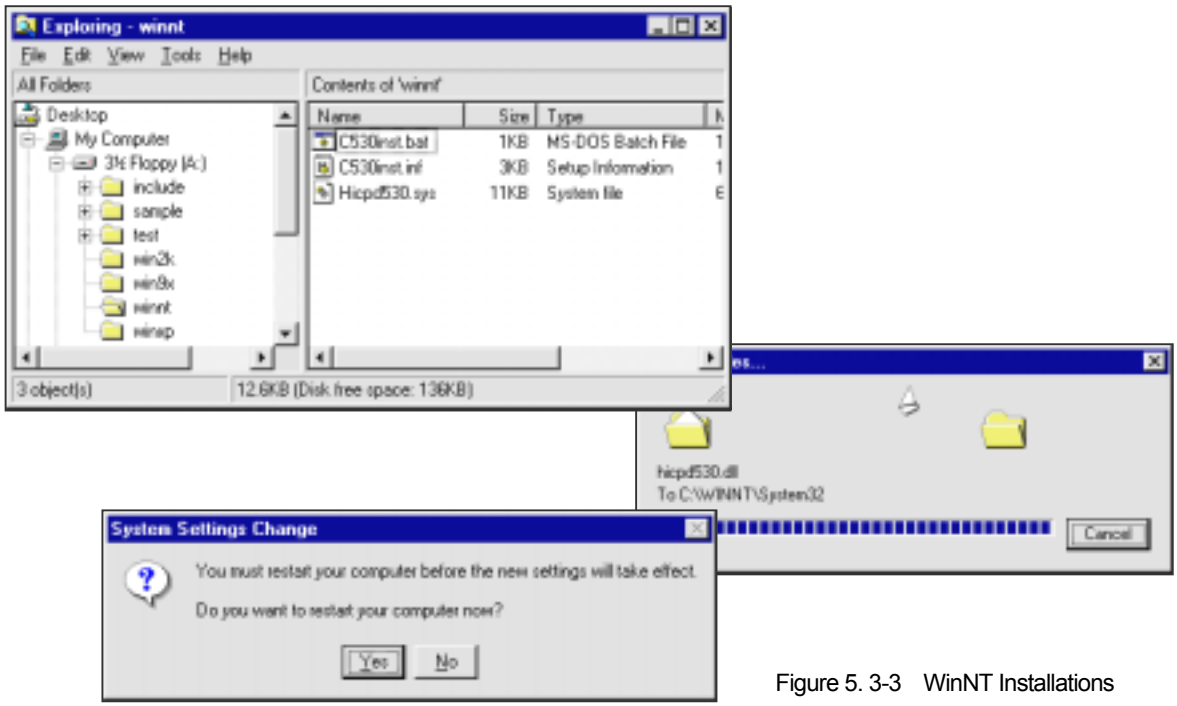


Figure 5. 3-3 WinNT Installations

[Starting and Stopping the Device]

After installing the device drivers, set the device driver to "automatic startup" so that the service for the CPD Board will be started when Windows NT starts up.
If the device must be stopped for some reason, use the following procedure to stop it.

1. Open the Control Panel and double-click the **Devices** icon. Select ***Hivertec HPCI-CPD530*** from the list of devices.
2. Click the **Startup** Button to display the Startup Type Button Box. Select **Disabled** for the Startup Type. (The device status will be "Stopped. ")
   To restart the CPD, open the Control Panel again and double-click the **Devices** icon. Select ***Hivertec HPCI-CPD530*** and select **Automatic** for the Startup Type. (The device status will be "Started. ")

(4) Windows 98 Installation

1. Verify that the computer's power supply is OFF and install the CPD Board in an available PCI slot in the computer. Turn ON the computer's power supply and start Windows.

2. When Win98 starts up, the CPD Board will be detected by the system and a window will be displayed automatically to guide you in the installation of the required device driver.
3. A dialog box will be displayed requesting the installation source directory. Insert the provided floppy disk into the computer's floppy disk drive.
4. Check the appropriate boxes to select the floppy disk drive as the search location.
5. Specify the directory "A:¥Win9x". (In this case, the floppy drive is drive A.)
   Complete the installation by following the directions in the system dialog boxes.

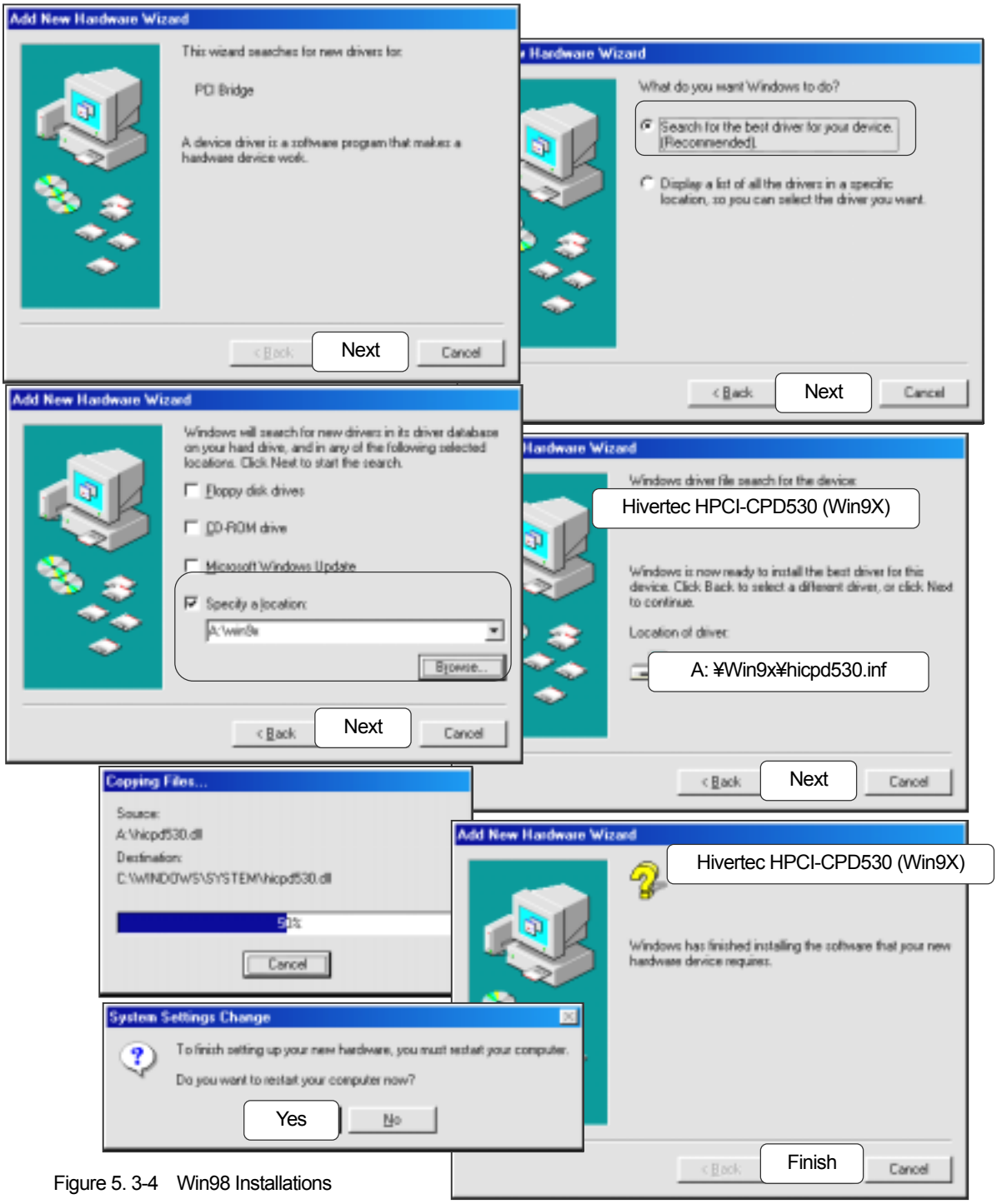

Add New Hardware Wizard
This wizard searches for new drivers for:
PCI Bridge
A device driver is a software program that makes a hardware device work.
Next
Cancel
Hardware Wizard
What do you want Windows to do?
Search for the best driver for your device. (Recommended).
Display a list of all the drivers in a specific location, so you can select the driver you want.
Back
Next
Cancel
Add New Hardware Wizard
Windows will search for new drivers in its driver database on your hard drive, and in any of the following selected locations. Click Next to start the search.
Floppy disk drives
CD-ROM drive
Microsoft Windows Update
Specify a location:
A:\win9x
Browse...
Back
Next
Cancel
Hardware Wizard
Windows driver file search for the device:
Hivertec HPCI-CPD530 (Win9X)
Windows is now ready to install the best driver for this device. Click Back to select a different driver, or click Next to continue.
Location of driver:
A: ¥Win9x¥hicpd530.inf
Back
Next
Cancel
Copying Files...
Source:
A:\hicpd530.dll
Destination:
C:\WINDOWS\SYSTEM\hicpd530.dll
50%
Cancel
Add New Hardware Wizard
Hivertec HPCI-CPD530 (Win9X)
Windows has finished installing the software that your new hardware device requires.
Back
Finish
Cancel
System Settings Change
To finish setting up your new hardware, you must restart your computer.
Do you want to restart your computer now?
Yes
No

Figure 5. 3-4 Win98 Installations

(5) Uninstalling the Device Driver for Windows

1. Insert the provided floppy disk into the computer's floppy disk drive.

2. Start Windows Explorer and execute the program "A:¥cp530uin. exe". (In this case, the floppy drive is drive A.)
It is also possible to start the device driver uninstallation program by executing A:¥cp530uin. exe from the command prompt.

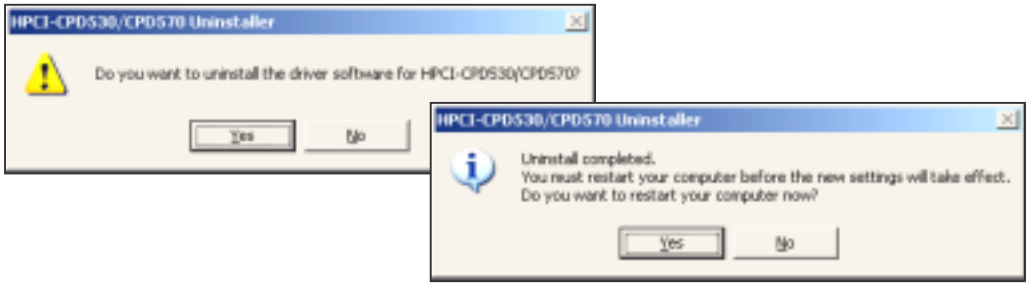

HPCI-CPD530/CPD570 Uninstaller
Do you want to uninstall the driver software for HPCI-CPD530/CPD570?
Yes
No
HPCI-CPD530/CPD570 Uninstaller
Uninstall completed.
You must restart your computer before the new settings will take effect.
Do you want to restart your computer now?
Yes
No

Figure 5. 3-5 Uninstalling the Device Drivers

5.3.2 Installing/Uninstalling Device Drivers in a DOS System

(1) DOS Installation

To register the HICPD530 drivers for the first time, copy the HICPD530. DRV file to the prescribed directory on the computer's hard disk and add the following line to the CONFIG. SYS file in the DOS startup drive. Specify the entire path in the DEVICE command. In this case, the file has been copied to the directory "C:¥HICPD".

DEVICE=C:¥HICPD¥HICPD530. DRV

Restart the computer after adding this line to the CONFIG. SYS file.

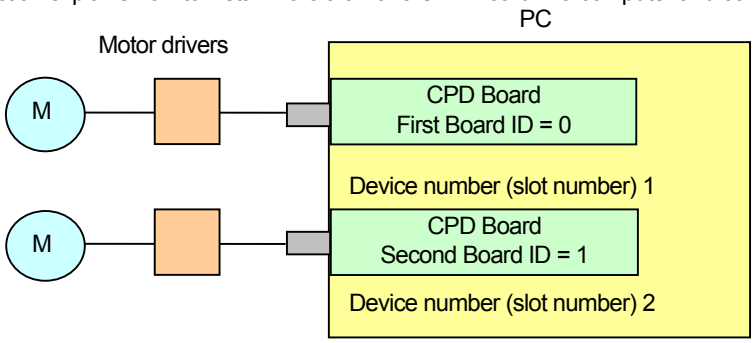
(2) Uninstalling the Driver

Locate the line that registers the HICPD530 drivers and delete this line from the "CONFIG. SYS" file.
(It is also possible to delete the driver file itself from the computer's hard disk.)

Restart the computer after deleting the appropriate line from the CONFIG. SYS file or deleting the HICPD530. DRV file itself.

5.4 Using Two or More CPD Boards

This section explains how to install more than one CPD Board in a computer and connect the Boards to external devices.



This example shows how to connect each Board to a separate driver.

Figure 5. 4-1 Using Multiple Boards

(1) Board Slot Number and Board ID

The computer's BIOS manage board addresses in the PCI bus system. The BIOS determines the slot number allocated to the slot in which the Board is mounted. The slot number is also known as the device number.

The slot numbers are dynamically allocated by the BIOS, so the relationship between the Boards and slot numbers cannot be tracked externally. Therefore, the CPD Boards are equipped with "Board ID Jumper Pins" to distinguish the Boards and coordinate the Board and software processing.

(2) Using Board IDs

The Board IDs can be set between 0 and 15. Up to sixteen CPD Boards (including CPD534, CPD532, CPD578, CPD508 Boards) can be controlled. Refer to *3. 3 Board Settings* for details on setting the Board ID.

5.5 Accessing the Board

It is possible to control more than one CPD Board with this function group. In order to access a particular CPD Board, it is necessary to open the device first and obtain the device handle value used to access the Board.
To open a device, it is necessary to know the hardware resources allocated to the device being opened. Hardware resources are resources such as the I/O port address and IRQ number, which are defined by the Windows or DOS system.

5.5.1 Data Structure for Board (Device) Recognition

(1) Windows Version of the Data Structure for Board (Device) Recognition

The following HPCDEVICEINFO structure is provided to recognize Boards. Use the number of structures required for the number of Boards being used (up to 16 Boards).

C language: Visual C++

```
typedef struct _HPCDEVICEINFO {
   DWORD      dwBusNumber;      /* Bus number */
   DWORD      dwDeviceNumber;   /* Device number */
   DWORD      dwIoPortAddress;  /* I/O port address */
   DWORD      dwIrqNo;          /* IRQ number */
   DWORD      dwNumber;         /* Management number */
   DWORD      dwBoardID;        /* Board ID (0 to 15) */
} HPCDEVICEINFO, *PHPCDEVICEINFO
```

Visual Basic

```
Public Type HPCDEVICEINFO
   dwBusNumber       As Long     ' Bus number
   dwDeviceNumber    As Long     ' Device number
   dwIoPortAddress   As Long     ' I/O port address
   dwIrqNo           As Long     ‘ IRQ number
   dwNumber          As Long     ' Management number
   dwBoardID         As Long     ' Board ID (0 to 15)
End Type
```

Note: The “Management number” is not used in Windows 98.
An “INVALID_HPC_NUMBER (-1)” will be stored.

(2) DOS Version of the Data Structure for Board (Device) Recognition

The following HPCDEVICEINFO structure is provided to recognize Boards. This structure is stored in the “hcpdtype. h” header file.

```
typedef struct   {
      WORD      nBusNumber;          /* Bus number */
      WORD      dwIoPortAddress;     /* I/O port address */
      WORD      dwIrqNo;             /* IRQ number */
      WORD      dwNumber;            /* Management number */
      WORD      dwBoardID;           /* Board ID */
}   HPCDEVICEINFO, *PHPCDEVICEINFO, far* LPHPCDEVICEINFO;
```

5.5.2 Board Access Preparatory Procedure and End Processing

(1) Using Driver Functions

When using the driver functions, the preparatory procedure and end processing are almost the same for both the Windows and DOS versions. Refer to the *CPD Board Series User's Manual: Software Information* for details on the driver functions.
When using interrupt processing in a DOS system, refer to the description of the DOS interrupt module registration function for details.

[Preparatory Procedure]

1. Obtain the device information for all of the Boards being used.
   First obtain the device information for all of the CPD Boards in the Board recognition structure area (array).
   cp530_GetDeviceCount() : Confirms the number of Boards.
   cp530_GetDeviceInfo() : Obtains the device information for all of the Boards.

2. Open every Board.
   Transfer the CPD Board's device information to the Open Device function.
   The CPD Board is opened and the Open Device function returns the device handle value that is needed to access that Board.
   When two or more Boards are being used in the computer, the processing must be performed for each individual Board.
   cp530_OpenDevice() : Opens the Board.

3. Make the initial settings for the Board's signal processing.
   From this point on, use the Board's device handle to access that CPD Board.
   Before initializing the Board, use the following function to set the Board's device operating conditions.
   Refer to the *CPD Board Series User's Manual: Software Information* for details on these settings.
   cp530_wPortB() : Sets each device's operating conditions.

4. Initialize each Board and each axis.
   After making the settings above, initialize each axis in all of the Boards being used.
   Refer to the User's Manuals and make the appropriate settings in each register.
   It will be possible to operate the axes normally once the settings have been made properly.

[End Processing]

5. Close the devices that were opened.
   When all of the required processing has been completed and the application has ended, close all of the devices that were opened.
   cp530_CloseDevice() : Closes the specified Board (1 Board).

(2) Using Library Functions

Refer to the *CPD Board Series User's Manual: Software Information* for details on the library functions.

[Preparatory Procedure]

1. Obtain the device information for all of the Boards being used.
   First obtain the device information for all of the CPD Boards in the Board recognition structure area (array).
   hcp530_GetDevInfo() : Confirms the number of Boards and obtains the device information for all of the boards.
   This library function combines two driver functions.

2. Open every Board.
   Transfer the CPD578 Board's device information to the Open Device function.
   The CPD Board is opened and the Open Device function returns the device handle value that is needed to access that Board.
   From this point on, use the Board's device handle to access that CPD Board.
   This step makes it possible to operate the axes normally, such as outputting pulses to each axis.
   When two or more Boards are being used in the computer, the processing must be performed for each individual Board.
   hcp530_DevOpen() :Opens the Board, makes the settings for the Board's signal processing, and sets each axis' operating conditions.
   This library function combines the driver functions in steps 2, 3, and 4 of the *Using Driver Functions* procedure.
   The initialization conditions are performed directly within the function. Refer to the *CPD Board Series User's Manual: Software Information* for more details on this function.

[End Processing]

3. Close the devices that were opened.
   When all of the required processing has been completed and the application has ended, close all of the devices that were opened.
   hcp530_DevClose() : Closes the specified Board (1 Board).

5.5.3 Verifying Each Axis is Ready for Operation

Verify the following points when each axis is connected to a motor and ready for operation.

1. Operational test of the ±ELS signals (The sensors are operated alone, with the motor stopped.)
2. Status of signal inputs when the servo alarm signals have been connected
3. Input status of the origin signal (OLS or Phase-Z)
4. Input status of the In-Position signal (Positioning completed: INPOS)

Proper operation may not be possible if the signals listed above are not being input properly.

The following points must be checked if the command pulse output to the motor is not operating properly.

1. Check that the command pulse output setting matches the “servo driver” input.
2. Check whether there is something in the “servo driver” input signal causing the motor to be stopped.

5.6 Sample Programs for Windows Systems

Sample programs have been provided to help explain how to use the library (level 1) functions. The following two kinds of programs have been provided and have almost identical displays and operations. The explanation in this section is based on the Visual C++ (5. 0) C coding sample program.

1. Visual C++ (5. 0) C coding (spc53000. exe)
2. Visual Basic (5. 0) (spc53002. exe)

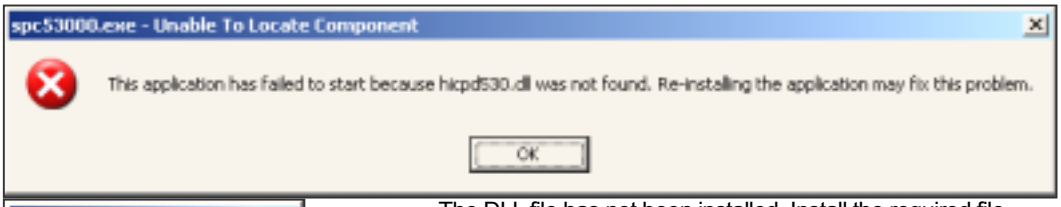
5.6.1 Executing the Sample Program

When using the sample program, copy the program to your hard disk.
The sample program can be executed by double-clicking the corresponding executable file (spc53000. exe or spc53002. exe).

(1) Precautions when Executing the Sample Program

1. When using the sample program for Visual C++, version 5. 0 (or higher) of Visual C++ must be installed already as the development tool.
2. When using the sample program for Visual Basic, version 5. 0 (or higher) of Visual Basic must be installed already as the development tool.
3. If the computer's OS is Windows 95 or Windows NT 4. 0 and Visual Basic 6. 0 is being used as the development tool, it may not be possible to execute spc53002. exe. In this case, spc53002. exe can be made executable by opening the project file "spc53002. vbp" and creating spc53002. exe again.
4. Set one CPD Board's Board ID to 0.
5. When two or more CPD Boards are being used, set a unique Board ID on each Board. If the same Board ID is set on more than one Board, the first Board found with that Board ID will operate.
6. The program will not run if one of the following error messages is displayed when you execute the program.

(2) Error Messages


spc53000.exe - Unable To Locate Component
This application has failed to start because hicpd530.dll was not found. Re-installing the application may fix this problem.
OK

The DLL file has not been installed. Install the required file.

A CPD530 Board is not installed or the Board is not recognized by the system. The device driver is not installed.

It is the mismatching of a system. Please ask a user support.

There isn't a Board installed with Board ID 0.

Figure 5. 6-1 Sample Program Error Messages

5.6.2 Sample Program Operations

There is a source program in the sample program that initializes each axis. Therefore, this source program must be changed if you want to operate the Board with different initialization conditions.

The following selection menu will be displayed if the sample program starts properly:

[Select Operation Menu]

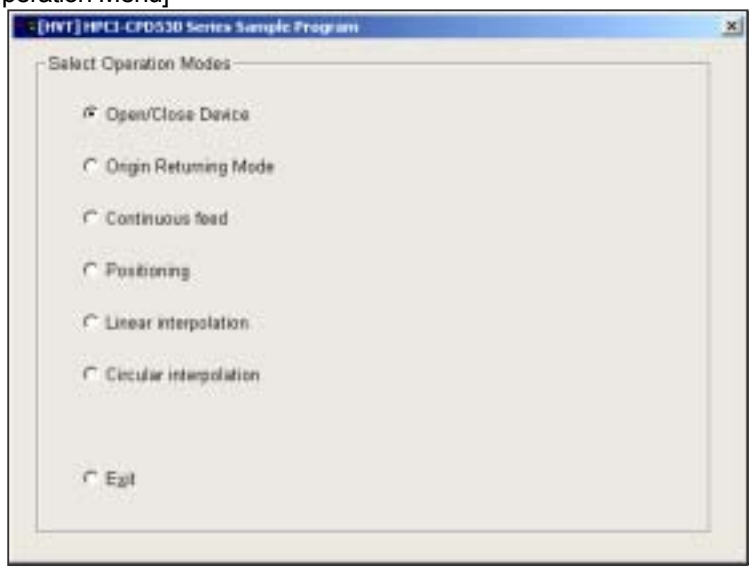


Figure 5. 6-2 The Sample Program's Select Operation Menu

Select the desired operation to execute that operation example.
(With the VC++ sample program, the operations are selected with a single-click. With the VB sample program, the operations are selected with a double-click.)

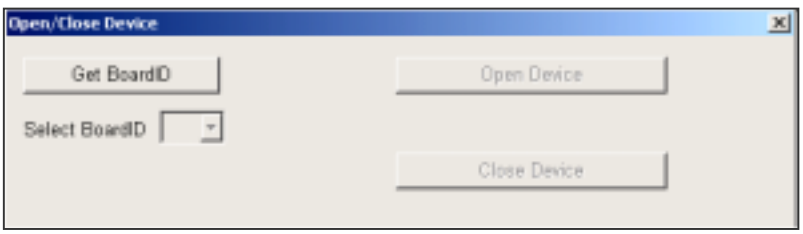
(1) Opening/Closing Devices

This sample program obtains device information and opens/closes the devices.
In order to access a CPD Board, it is necessary to first open the device and obtain the device handle that is required to access the device. The Open Device function initializes each register and the option ports at the same time that it opens the device and obtains the device handle.

In this sample operation, its Board ID selects the Board, the device information is obtained for the selected Board, and that Board's device is opened.
Also, that Board's device is closed with the Close Device function.
For more details on device information and opening/closing devices, refer to the *CPD Board Series User's Manual: Common Information*.

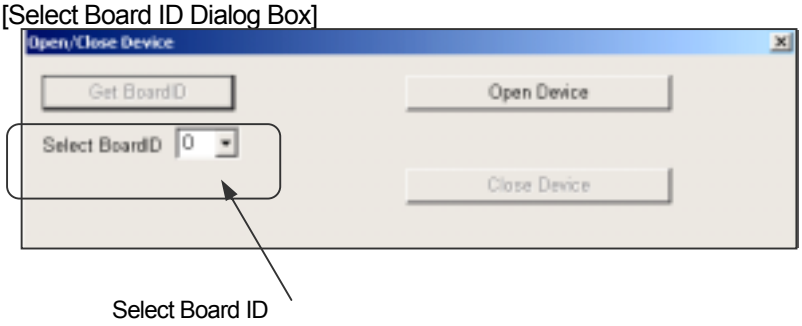
The following diagrams show how to perform the sample operations.

[Device Opening/Closing Dialog Box]



When you click "**Get Board ID"** in the dialog box, the **"Select Board ID"** area will display the Board IDs of the Boards that are actually installed in the computer.

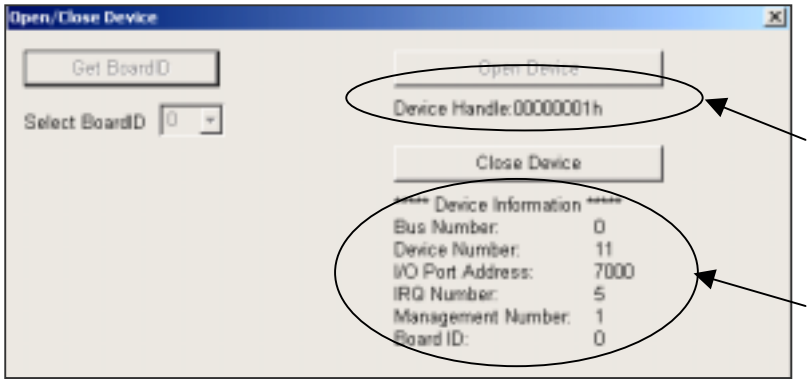
[Select Board ID Dialog Box]



Select Board ID

Select the desired Board ID and click the **"Open Device"** Button to open the device.

[Window after Open Device Button is clicked]



Device handle obtained by Open Device

Device information for the device that was opened

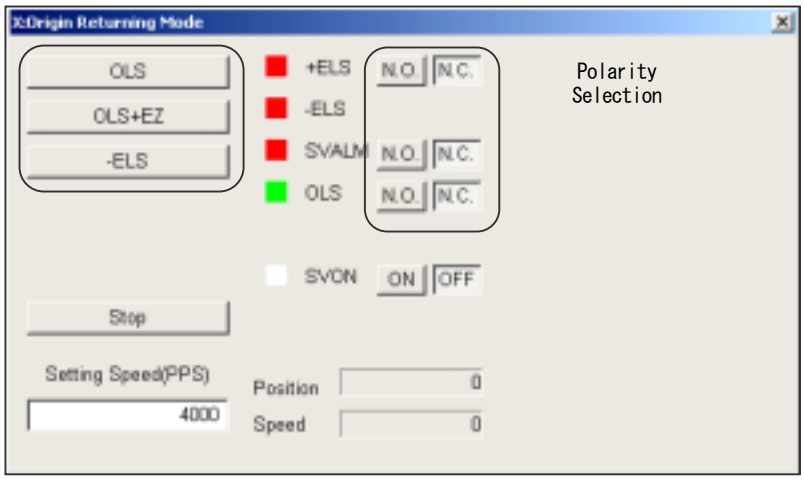
Click the **"Close Device"** Button to close the device and return to the **"Select Board ID"** Dialog Box.

(2) Origin Returning Operation

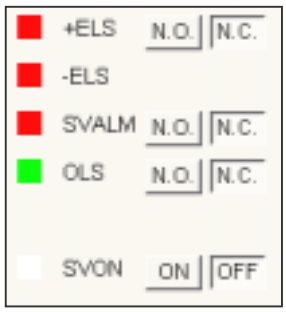
This sample program makes the settings for the origin returning operation and performs the origin returning operation.

[Initial Window for the Origin Returning Operation]

Select Origin Returning Operation



[Preparations for Operation]

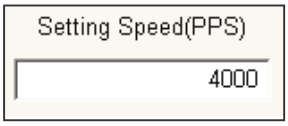


1. Polarity Selection

The indicated color will change when there is an input from the sensor. The +ELS, -ELS, and SVALM indicators will be red when there is an input and the OLS indicator will be green.
If the SVON signal is being output, the indicator will be green.
The input polarity can be switched by clicking the **"N. O. "** and **"N. C. "** Buttons.
The Servo can be turned ON and OFF by clicking the **"ON"** and **"OFF"** Buttons.

When a stepping motor driver is being used, the **"OFF"** Button turns the excitation ON and the **"ON"** Button turns the excitation OFF.

Note 1. Operation is disabled while +ELS, -ELS, or SVALM is being input. Start operation after checking the status of each sensor.
It is assumed that SVON is connected as specified.
Note 2. With a normally open (N. O.) terminal, the signal is ON (detected) when there is a current.
With a normally closed (N. C.) terminal, there is normally a current flowing and the signal is ON (detected) when there is no current.



2. Setting the Operation Speed

The operation speed can be set in the range of 1 to 65,535 pps. The initial value is 4,000 pps, so change this initial value to the desired speed when necessary.
Also, the base speed is set to 400 pps. If the operation speed is set below 400 pps, the motor will accelerate when the speed drops to 400 pps even if it is supposed to decelerate because the DLS or OLS signal is ON. In this case, change the sample program and set the base speed to the appropriate value.

[Executing the Origin Returning Operation]

Any one of the following origin returning methods can be selected.
"OLS": Origin returning operation 1(After OLS is detected, Get out of origin, re-enter and end.)
"OLS+EZ": Origin returning operation 2: OLS ON is detected and encoder phase-Z is detected.
" ELS": Origin returning operation 6: Reverse when ELS is detected and end when ELS is exited.
Refer to the *CPD Board Series User's Manual: Software Information* for more details on the origin returning operations.

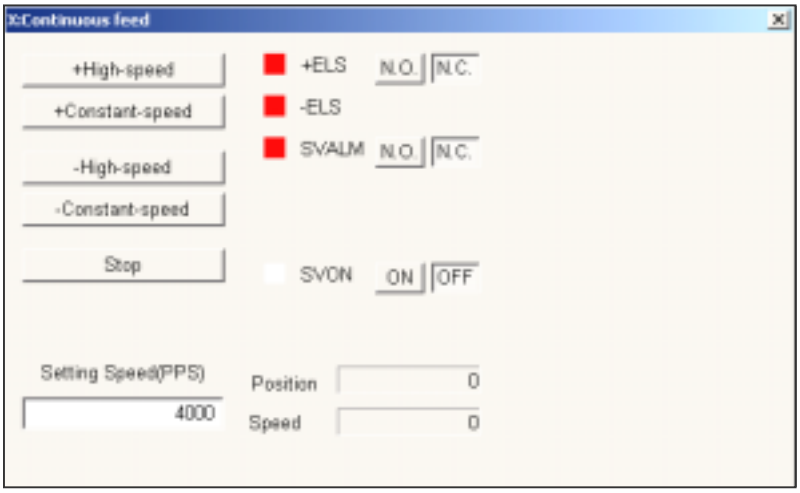
After completing, the "Preparations for Operation" and specifying one of the origin returning methods, click the **"Origin Returning"** Button to execute the specified origin returning operation.
Clicking the **"Stop"** Button can stop the operation.
The current position display indicates the command pulse count.
The current speed display indicates the pulse speed (pps) currently being output.

Note The OLS signal detection is rising-edge detection, so the OLS signal will not be detected if the signal is already ON when operation starts. In this case, perform continuous feed operation until the OLS signal goes OFF and then execute the origin returning operation.

(3) Continuous Feed Operation

This sample program demonstrates the high-speed continuous feed and constant-speed continuous feed operations.

[Continuous Feed Operation Window]



Before starting operation, perform the same preparatory checks (such as checking sensor connections) performed for the origin returning operation.

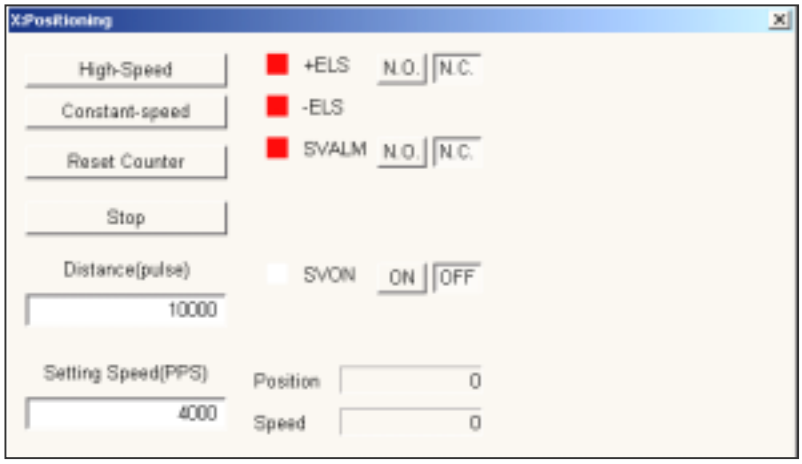
Click the **"+High-speed"**, **"+Constant-speed"**, **"-High-speed"**, or **"-Constant-speed"** Button to perform the corresponding continuous feed operation.

Clicking the **"Stop"** Button can stop the operation.

(4) Positioning Operation

This sample program demonstrates the high-speed positioning and constant-speed positioning operations.

[Positioning Operation Window]



Before starting operation, perform the same preparatory checks (such as checking sensor connections) performed for the origin returning operation.

Click the **"High-speed"**or **"Constant-speed"** Button to perform the corresponding continuous feed operation.

Set the distance in pulse units.

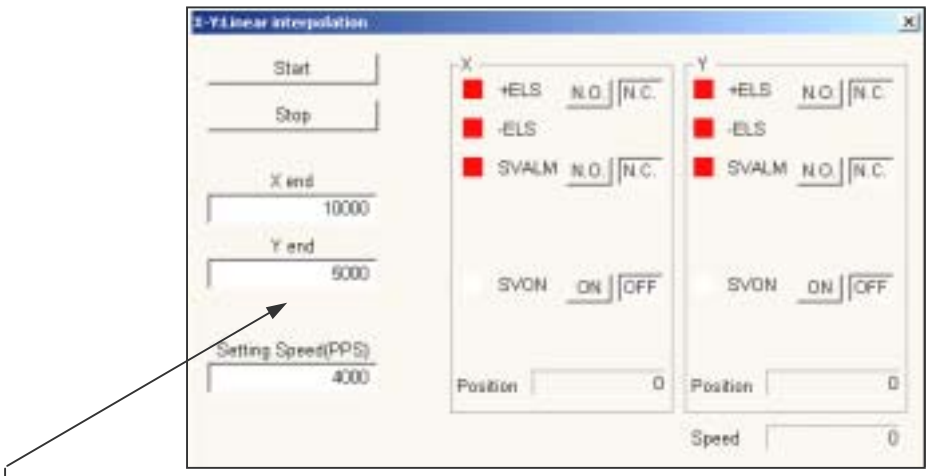
The current position can be reset to 0 by clicking the **"Reset Counter"** Button.

Clicking the **"Stop"** Button can stop the operation.

(5) Linear Interpolation Operation

This sample program performs linear interpolation operation at high speed. The combined speed is fixed.

[Linear Interpolation Operation Window]



Before starting operation, perform the same preparatory checks (such as checking sensor connections) performed for the origin returning operation.

Set the end positions for the X-axis and Y-axis.

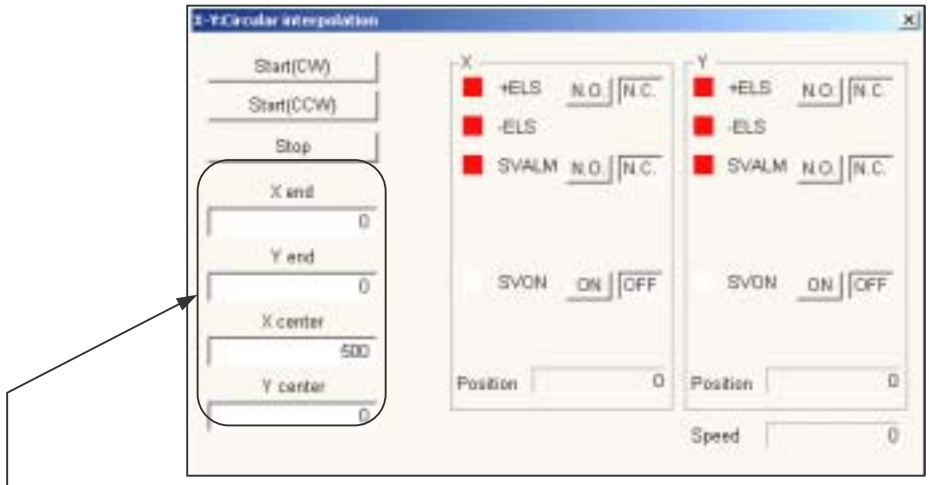
Click the **"Start"** Button to perform the linear interpolation operation.

Clicking the **"Stop"** Button can stop the operation.

(6) Circular Interpolation Operation

This sample program performs circular interpolation operation at the base speed. (Constant surface speed control with a base speed of 500 pps)

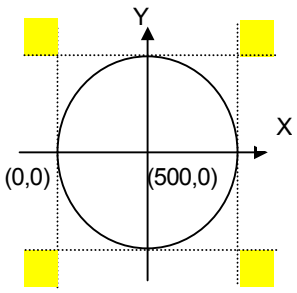
[Circular Interpolation Operation Window]



Before starting operation, perform the same preparatory checks (such as checking sensor connections) performed for the origin returning operation.

Set the end position and center position for the X-axis and Y-axis.

Note The current point is the starting point, and the end position is at the end coordinate values relative to that starting point.
When the end point value is set to (0,0), a complete circle will be formed.
If the end point coordinates are not on the circle, end point withdrawal will start when the operation reaches either the specified X-axis or Y-axis end position. If the end position has been set in one of the yellow regions shown in the diagram on the right, the operation will never reach the X-axis or Y-axis end position so operation will not stop.



Setting example

Click the **"Start (CW)" or "Start (CCW)"** Button to perform the circular Interpolation operation.

Clicking the "Stop" Button can stop the operation.

5.7 ”Let’s try to move” Program for Windows Systems

The “Let’s try to move” program software allows you to check minimal operations on the display by simply connecting Board to computer. Execute the A:¥test¥Release¥tpc53000. exe file on the provided floppy disk. (In this case, the floppy disk drive is drive A.)

< Caution >

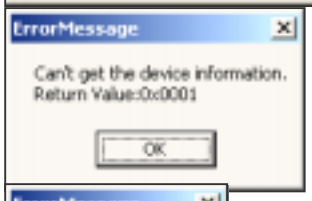
When two or more CPD Boards are being used, set a unique Board ID on each Board. If the same Board ID is set on more than one Board, the first Board found with that Board ID will operate.

For safety reasons, the display will not change with this application while the axis is operating.

The program will not run if one of the following error messages is displayed when you execute the program.

[Error Messages]

The DLL file has not been installed.
The installation is incomplete.



A CPD530 Board is not installed or the Board is not recognized by the system.
The device driver is not installed.

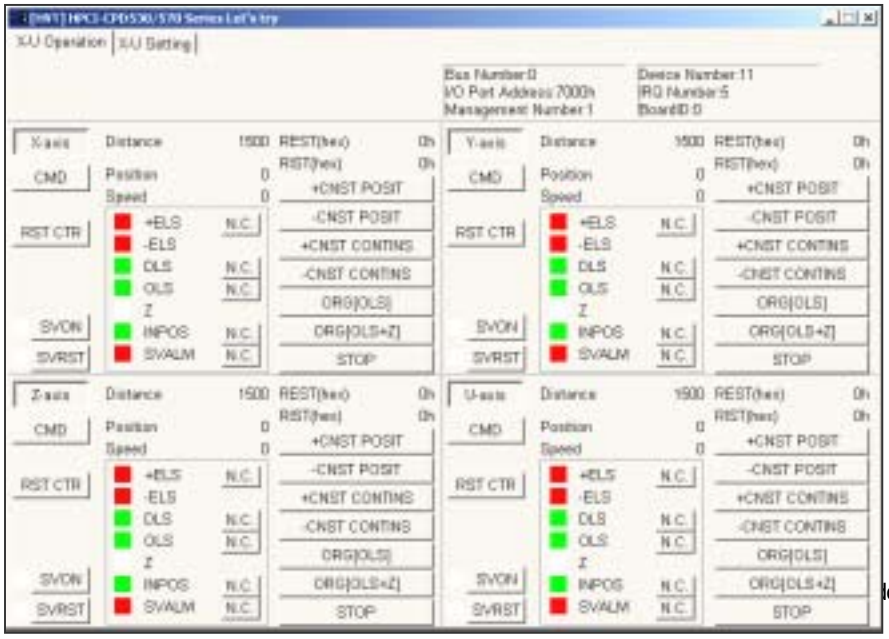
It is the mismatching of a system. Please ask a user support.

Figure 5. 7-1 Let’s try to move Error Messages

5.7.1 ”Let’s try to move” Operation Window

The following window will be shown when the “Let’s try to move” program is executed. When the program starts, it will operate with the first Board that is found. To operate the “Let’s try to move” program with a different Board, select the Setting Window by clicking the **“X-U Setting”** tab at the top of the Window.

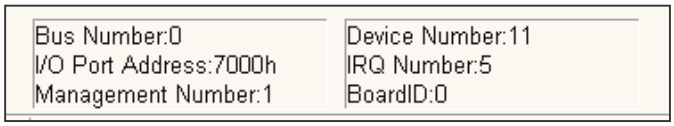
Refer to *5. 7. 2 Let’s try to move Settings Window* for details on selecting another Board.



low

(1) Device Information Display

The device information for the Board that is currently selected is displayed near the top of the window, as shown below.



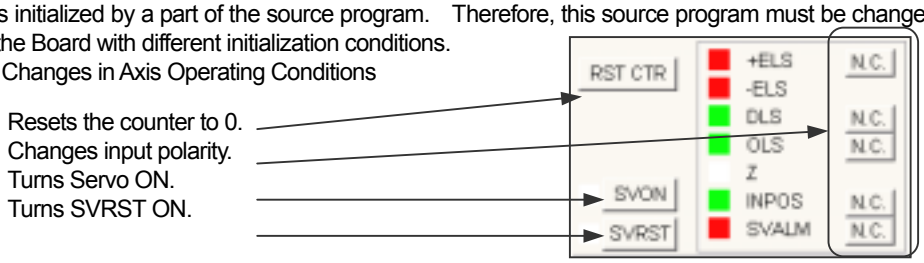
Note 1.The “NTLP: Management Number” will be “-1” in a Win9x or WinMe system.
Note 2.The “Board ID” is the value set with the Board’s Board ID jumper pins.
Note 3.The “Device number” (The slot number)

(2) Individual Axis Displays and Operation Commands

The operations are the same for each axis on the Board.

Each axis is initialized by a part of the source program. Therefore, this source program must be changed if you want to operate the Board with different initialization conditions.

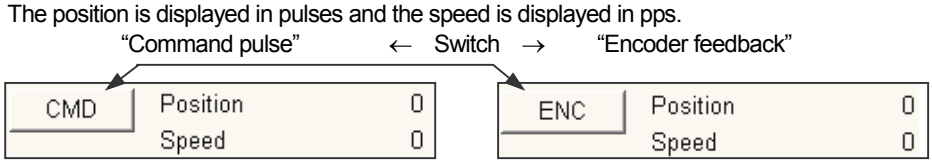
1. Axis Status and Changes in Axis Operating Conditions



2. Axis’ Current Position and Operation Speed

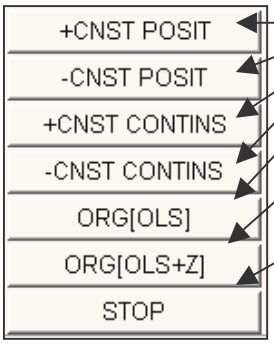
Each axis’ current position and NTLP:operation speed is refreshed approximately every 0. 1 s.
The current position display can be switched between the command-output-pulse and the encoder-feedback.

The position is displayed in pulses and the speed is displayed in pps.



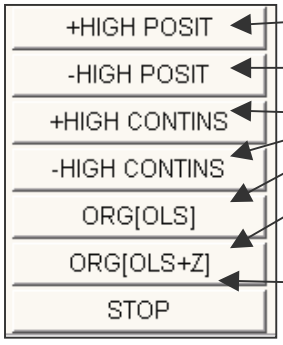
3. Sending Start and Stop Commands to an Axis. Each axis' operation can be set to constant/high-speed positioning operation, constant/high-speed continuous feed operation, constant/high-speed origin returning operation, or Stop. Refer to *5. 7. 2 Let's try to move Settings Window* for details on the axis operating conditions that can be changed.

Setting Constant-speed Operation



1. Positioning in "+" direction at base speed. (Set positioning amount in Settings Window.)
2. Positioning in "–" direction at base speed. (Set positioning amount in Settings Window.)
3. Constant-speed operation in "+" direction at base speed.
4. Constant-speed operation in "–" direction at base speed.
5. Origin returning operation in "–" direction at base speed. When the OLS signal goes ON, operation stops immediately and the origin returning operation is completed.
6. High-speed origin returning operation in "–" direction at operation speed. When the OLS signal goes ON, operation decelerates to base speed. When the first Z-phase input is received, operation stops immediately and the origin returning operation is completed.
7. When operating at operation speed, operation is decelerated to a stop.
   When operating at base speed, operation is stopped immediately.

Setting High-speed Operation



1. High-speed positioning in "+" direction at operation speed.
   (Set positioning amount in Settings Window.)
2. High-speed positioning in "–" direction at operation speed.
   (Set positioning amount in Settings Window.)
3. High-speed operation in "+" direction at operation speed.
4. High-speed operation in "–" direction at operation speed.
5. Origin returning operation in "–" direction at base speed. When the OLS signal goes ON, operation stops immediately and the origin returning operation is completed.
6. High-speed origin returning operation in "–" direction at operation speed. When the OLS signal goes ON, operation decelerates to base speed. When the first Z-phase input is received, operation stops immediately and the origin returning operation is completed.
7. When operating at operation speed, operation is decelerated to a stop.
   When operating at base speed, operation is stopped immediately.

Note 1. Acceleration/deceleration operations use linear acceleration/deceleration.
Note 2. The DLS signal is enabled. If this signal is not being used, set it as an N. O. condition and set it so that it is not input.
Note 3. The INPOS signal is enabled. If this signal is not being used, set it as an N. C. condition and set it so that it is not normally input.
Note 4. The OLS signal detection is rising-edge detection, so the OLS signal will not be detected if the signal is already ON when operation starts. In this case, perform continuous feed operation until the OLS signal goes OFF and then execute the origin returning operation.
Note 5. During operation, if EA/EB error or PA/PB error occur, the"POSIT"Button, the"CONTINS"Button, and the"ORG"Button is enabled.

5.7.2 ”Let’s try to move” Settings Window

The following settings window will be displayed when all axes were stopped in the Let’s try to move Operation Confirmation Window and the **“X-U Setting”** tab was clicked at the top of that Window.

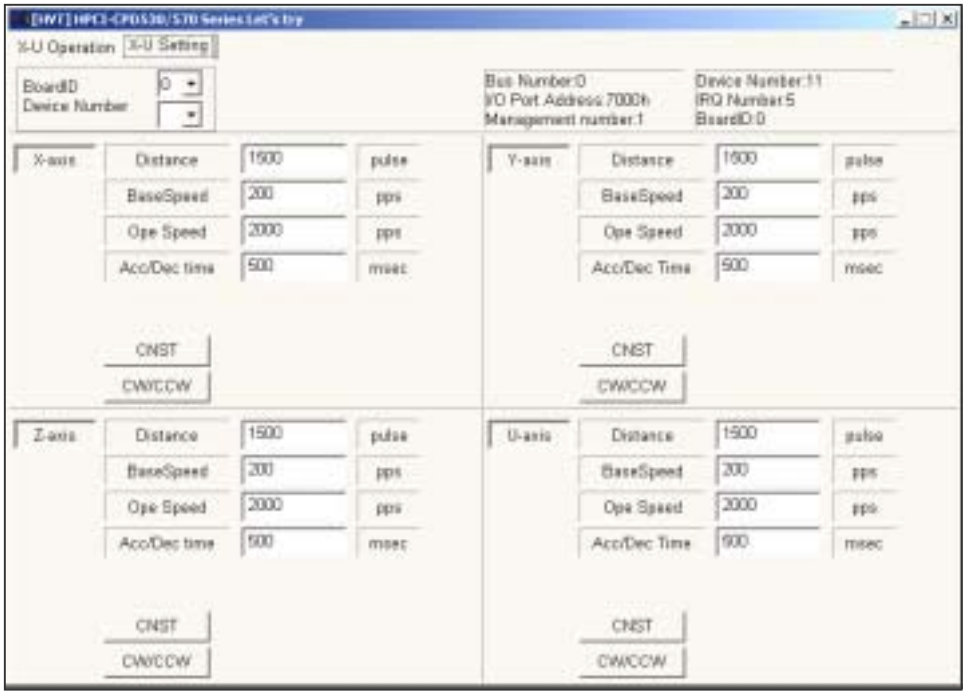
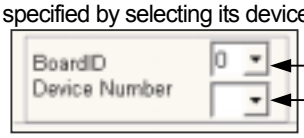


Figure 5. 7-3 Let’s try to move Settings Window

(1) Board Selection and Device Information

When two or more Boards have been installed in the computer, the Board used for the Let’s try to move can be specified by selecting its device number or Board ID, as shown below.

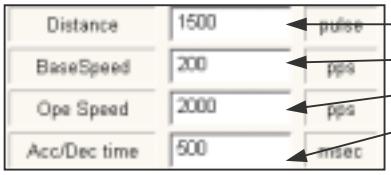


1. Select by Board ID.
2. Select by device number.

(2) Changeable Axis Operating Conditions

The operating conditions can be set individually for each axis that can operate.

1. Switches between constant-speed and high-speed operation. The default setting is constant-speed.
2. The command pulse output format can be switched between discrete and shared pulse methods. The default setting is the discrete pulse method.



1. Sets the travel distance. (1,500)
2. Sets the base speed. (200)
3. Sets the operation speed. (2,000)
4. Sets the acceleration/deceleration time. (500)

Note. The default settings are in parentheses.

Note 1. The setting range for the travel distance is –999,999 to +9,999,999 pulses.

Note 2. The setting ranges for both the base speed and operation speed are 1 to 65,535 pps, although the full setting range may not be usable depending on the combination of settings being used. For the same reason, it may not be possible to use the full setting range of the acceleration/deceleration time depending on the settings made for the operation speed and base speed.

Note 3. If the operation speed is set below the base speed, the motor will accelerate when the speed drops to 400 pps even if it is supposed to decelerate because the Stop command is input, the DLS is ON, or the OLS signal is ON.

Note 4. Error Message

The window shown on the right will be displayed if an invalid setting has been input. Click **OK** and enter a valid setting.

5.8 Sample Programs for a DOS System

Sample programs written in C language have been bundled with the DOS-version software. Use the sample programs for the following purposes.

(1) Checking the Installed Boards

After installing the device driver, turning OFF the computer's power supply, installing the Boards, and restarting the computer, the sample program's executable file can be started to display the mounted Boards' "device information" and confirming the Boards' operation.

(2) Demonstrating the Functions in the Driver Interface Library

The application program must perform various operations on the Board through the driver interface library and device driver. The sample program provides examples of the operations.

5.8.1 Configuration of the Sample Program

The sample program and source files have the following configuration.

```
clk. bat              --- Batch file that creates executable files for MS-C (Ver.  6. 0)
 |------    smp530. c    ----Source program
 |------    hicpd530. h  ---- Header file    "#include" by "smp530. c   "
 |------    hcpdtype. h  ---- Header file    "#include" by "hicpd530. h"
 |-------   licpd530. lib ---- Library file (the initial "l" indicates large model)
```

5.8.2 Starting the Sample Program

The following displays will appear when the executable "smp530. exe" file is started.

(1) Sample Program Terminated by Error

The following error message will be displayed if the device driver has not been installed in the system or the Board has not been installed in the computer.

```
 *** HPCI-CPD532,534,508,578A : Device Driver Sample [ Board =  0 ] ***
    ----- hicpd530.lib Ver0.00  &  hicpd530.drv Ver0.00 -- [ non board ]

     ***  End of Sample Program  ***
C:¥cpd530¥SMP>
```

If this message is displayed, check whether the device driver and Board have been installed properly.

(2) Sample Program Started Normally

The following message will be displayed if the program started normally.

```
 *** HPCI-CPD532,534,508,578A : Device Driver Sample [ Board =  1 ] ***
    ----- hicpd530.lib Ver1.10  &  hicpd530.drv Ver1.10 -----

                   Bus= 0 Dev=12  I/O=0x7000 IRQ=11 CTL= 1  BID= 0  Open?=_
```

The Board displayed here can be selected by pressing one key.
To select operation with the interrupt function, input "I" or "i". To select operation without the interrupt function, input "Y" or "y". If any other key is pressed, the displayed Board will not be selected (when 2 or more Boards are installed).

The following diagrams show the different screens displayed when "y" and "I" are input.
In the screen displayed after "y" is pressed, the Board's interrupt function is not used. In the screen displayed after "I" is pressed, the Board's interrupt function is used.

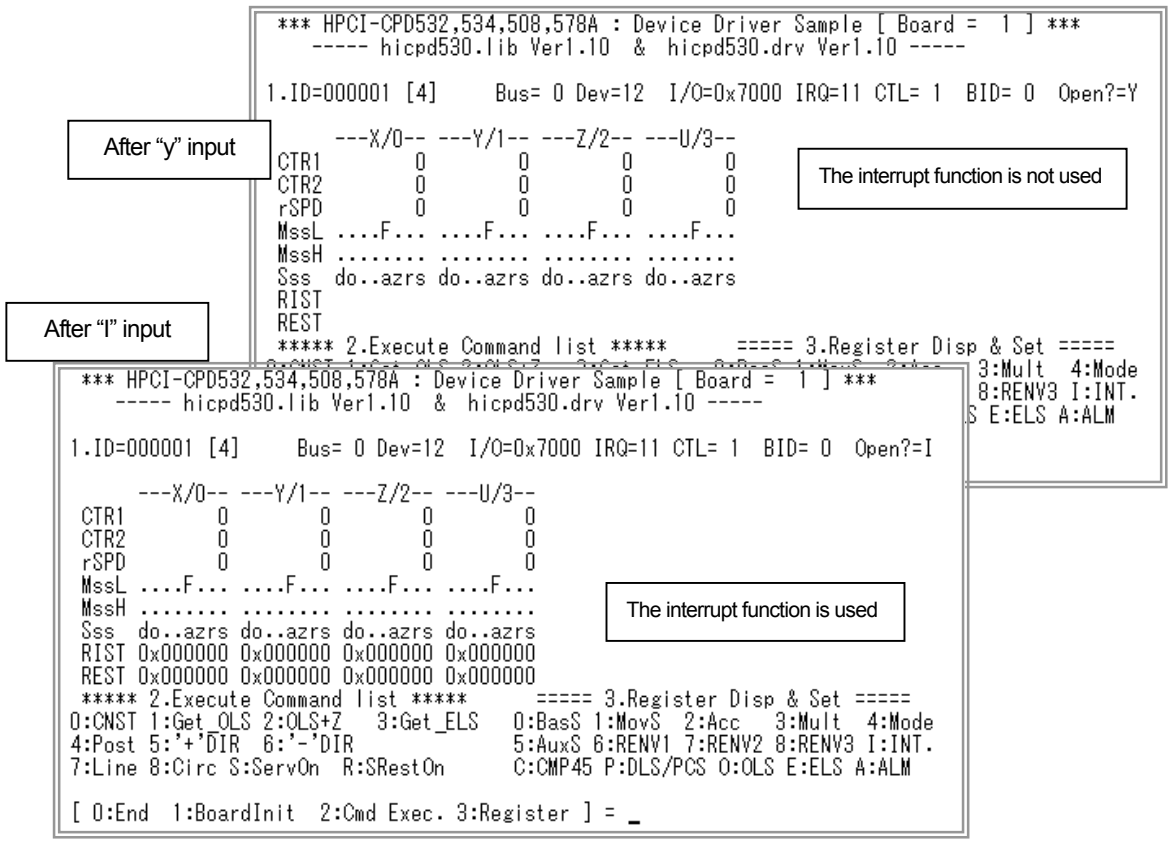
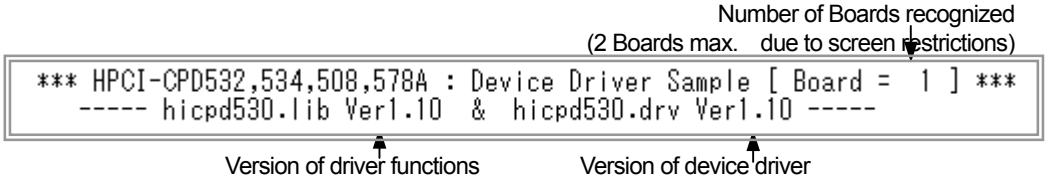

*** HPCI-CPD532,534,508,578A : Device Driver Sample [Board = 1] ***
----- hicpd530.lib Ver1.10 & hicpd530.drv Ver1.10 -----
1.ID=000001 [4] Bus= 0 Dev=12 I/O=0x7000 IRQ=11 CTL= 1 BID= 0 Open?=Y
---X/0-- ---Y/1-- ---Z/2-- ---U/3--
CTR1 0 0 0 0
CTR2 0 0 0 0
rSPD 0 0 0 0
MssL ....F... ....F... ....F... ....F...
MssH ....
Sss do..azrs do..azrs do..azrs do..azrs
RIST
REST
***** 2.Execute Command list ***** ===== 3.Register Disp & Set =====
After "y" input
The interrupt function is not used
After "I" input
*** HPCI-CPD532,534,508,578A : Device Driver Sample [Board = 1] ***
----- hicpd530.lib Ver1.10 & hicpd530.drv Ver1.10 -----
1.ID=000001 [4] Bus= 0 Dev=12 I/O=0x7000 IRQ=11 CTL= 1 BID= 0 Open?=I
---X/0-- ---Y/1-- ---Z/2-- ---U/3--
CTR1 0 0 0 0
CTR2 0 0 0 0
rSPD 0 0 0 0
MssL ....F... ....F... ....F... ....F...
MssH ....
Sss do..azrs do..azrs do..azrs do..azrs
RIST 0x000000 0x000000 0x000000 0x000000
REST 0x000000 0x000000 0x000000 0x000000
***** 2.Execute Command list ***** ===== 3.Register Disp & Set =====
0:CNST 1:Get_OLS 2:OLS+Z 3:Get_ELS 0:BasS 1:MovS 2:Acc 3:Mult 4:Mode
4:Post 5:'+'DIR 6:'-'DIR 5:AuxS 6:RENV1 7:RENV2 8:RENV3 I:INT.
7:Line 8:Circ S:ServOn R:SRestOn C:CMP45 P:DLS/PCS O:OLS E:ELS A:ALM
[0:End 1:BoardInit 2:Cmd Exec. 3:Register] = _
The interrupt function is used

Figure 5. 8-1 Startup Screens for the DOS Version of the Sample Program

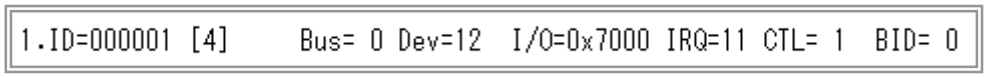
The differences in the "interrupt display" for these two screens are as follows:
The relevant interrupt displays are in the Event Status (RIST) and Error Status (REST) fields.

* Interrupts not used:
  Interrupt sources are polled. When an interrupt occurs, the source is displayed in hexadecimal for a fixed time.
* Interrupts used:
  When interrupts have occurred, the interrupt sources are displayed in hexadecimal as OR's.
  (The display is cleared at the start of operation.)

(1) Title, Number of Boards Recognized, and Software Version


Number of Boards recognized
(2 Boards max. due to screen restrictions)
*** HPCI-CPD532,534,508,578A : Device Driver Sample [Board = 1] ***
----- hicpd530.lib Ver1.10 & hicpd530.drv Ver1.10 -----
Version of driver functions
Version of device driver

(2) Board Information



(a) Display of Selected Boards (Operable Boards)

ID: This hexadecimal ID is the device handle returned to the Device Open function.
The Board ID (BID) is added to the higher word of the device ID in the Device Open function
[N]: Displays the number of enabled axes (N) that were recognized in the sample program.

(b) Board Display

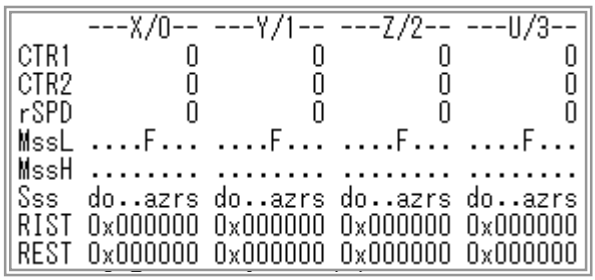
Indicates the following device information that was obtained for the selected Board.

Bus:	Bus number	Dev:	Device number (PCI slot number)
I/O:	I/O port address	IRQ:	IRQ number
CTL:	Management number	BID:	Board ID (0 to 15, set with the Board's jumper pins)

(3) Board Status (Selected Board Only)

The following information is displayed for each axis.

CTR1: Current command pulse position
CTR2: Current feedback position
rSPD: Operating speed (0 to 16,383)
MssL: Main status: Lower 8 bits
MssH: Main status: Higher 8 bits
Sss Sub-status + Encoder phase-Z
RIST: Event status (interrupt source)
REST: Error status (interrupt source)

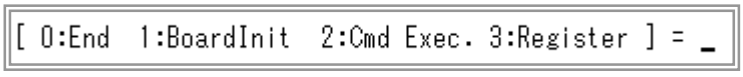


The status display is expressed in one character.

MssL	Bit	b7	b6	b5	b4	b3	b2	b1	b0
	Name	SSC1	SSC0	SINT	SERR	SEND	---	SRUN	SSCM
	Display 1	**o**	**o**	**I**	**E**	**F**	**.**	**R**	**C**
	Display 0	**.**	**.**	**.**	**.**	**.**	**.**	**.**	**.**
MssH	Bit	b15	b14	b13	b12	b11	b10	b9	b8
	Name	SPDF	SPRF	---	SCMP5	SCMP4	SCMP3	SCMP2	SCMP1
	Display 1	**5**	**P**	**.**	**5**	**4**	**3**	**2**	**1**
	Display 0	**.**	**.**	**.**	**.**	**.**	**.**	**.**	**.**
Sss (RSTS)	Bit	b15	b14	b13	b12	b11	RSTS:b10	b1	b0
	Name	SDLS	SOLS	SMEL	SPEL	SALM	SEZ	SVRST	SVON
	Display 1	**D**	**O**	**-**	**+**	**A**	**Z**	**R**	**S**
	Display 0	**d**	**o**	**.**	**.**	**a**	**z**	**r**	**s**

Note: RSTS = Expansion status register and b10/SEZ = EZ input signal

(4) Selecting Program Commands



Input one of the following characters from the keyboard to select a command.

0: End the program
1: Initialize the specified Board. (The Board is initialized when the program starts up.)
2: Send an operation command to the specified axes on the specified board.
3: Set the register values for the axis LSI registers. (Shared by all axes.)
Other: If any other key is input, all axes will be decelerated to a stop.

(5) Using Operation Commands and the Main Status

The sample program does not confirm the startup conditions of commands to individual axes or the end of those operations. The typical application program controls continuous independent operation to individual axes or interpolation operation to specified axes.

At that time, it is necessary to confirm whether the operation is performed properly and the operation's end status.

The main status register (MSTS) is provided for this purpose. Refer to *6. 2 Using the Main Status Register* for details.

5.8.3 Sample Program Operations

Use the following procedure to perform various operations in the sample program.

(1) Register Contents Table

This table contains the contents of registers that are shared by the axes and can be changed. The set values of individual registers can be displayed by selecting **3:Register** from the program commands. The set values can be changed and written to the axes with keyboard inputs.

```
  ===== 3.Register Disp & Set =====
0:BasS 1:MovS  2:Acc   3:Mult  4:Mode
5:AuxS 6:RENV1 7:RENV2 8:RENV3 I:INT.
C:CMP45 P:DLS/PCS O:OLS E:ELS A:ALM
```

(2) Displaying and Changing Registers

Name	Key input	Contents	Name	Key input	Contents
0:BasS	0	Base speed	I:INT.	I	Event mask
1:MovS	1	Operation speed	C:CMP45	C	Comparators 4 and 5
2:Acc	2	Acceleration rate	P:DLS/PCS	P	DLS/PCS
3:Mult	3	Speed multiplier	O:OLS	O	OLS input polarity invert
4:Mode	4	Operation mode	E:ELS	E	ELS input polarity invert
5:AuxS	5	Travel distance compensation speed	A:ALM	A	ALM input polarity invert
6:RENV1	6	Environment 1			
7:RENV2	7	Environment 2			
8:RENV3	8	Environment 3			

The table above lists the 15 relevant items in the register.
Select **3:Register** and then input the character corresponding to the item that you want to display/change.

Input the desired value into this field to set the new value in all axes. If you press just the **Enter** Key alone, the operation will end without changing the setting for the axes.
When one of the 3 kinds of input sensors is selected, the input polarity for all axes will be inverted immediately and the operation will end. The results of this "input polarity invert "operation will be reflected immediately in the display.

The ELS of all axes detected in both directions.

```
MssL ....F... ....F... ....F... ....F...
MssH ........ ........ ........ ........
Sss  do-+azrs do-+azrs do-+azrs do-+azrs
```

```
[ 0:End  1:BoardInit  2:Cmd Exec. 3:Register ] = 3
 Select Reg( 0-8,I,C,P,O,E,A ) = E

   ELS[N.Open  to N.Close]
```

As the ELS input polarity is inverted for all axes, so the ELS status changes to undetected.

```
MssL ....F... ....F... ....F... ....F...
MssH ........ ........ ........ ........
Sss  do..azrs do..azrs do..azrs do..azrs
```

(3) Initializing the Board

This display initialized the 4-axis Board.
In the initialization operation, the following commands are executed for the axes in the specified Board and then all registers are reset to their initial values.

(a) Perform Software Reset on each axis.
(b) Set data in registers.
Write settings in the register contents table. Write fixed values (0) to the registers that are not in the table.

```
     ---X/0-- ---Y/1-- ---Z/2-- ---U/3--
 CTR1        0        0        0        0
 CTR2        0        0        0        0
 rSPD        0        0        0        0
 MssL ....F... ....F... ....F... ....F...
 MssH ........ ........ ........ ........
 Sss  do-+azrs do-+azrs do-+azrs do-+azrs
 RIST
 REST
 ***** 2.Execute Command list *****      ===== 3.Re
0:CNST 1:Get_OLS 2:OLS+Z   3:Get_ELS   0:BasS 1:Mov
4:Post 5:'+'DIR  6:'-'DIR              5:AuxS 6:REN
7:Line 8:Circ S:ServOn  R:SRestOn      C:CMP45 P:DL

[ 0:End  1:BoardInit  2:Cmd Exec. 3:Register ] = 1
```

(4) Sending Operation Commands to Each Axis

There are 11 kinds of operation commands, as shown in the table on the right, but these commands can be divided into 3 categories.

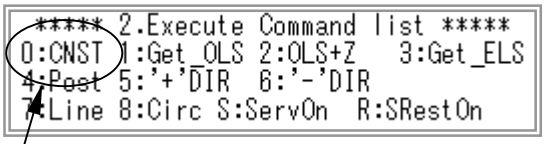
* Change operation speed:---- 0: Constant speed

*Servo signal output to all axes:--- S: Servo ON

---R: Reset ON

* Operation Command to individual axis:

All inputs other than 0, S, and R



(a) Change operation speed

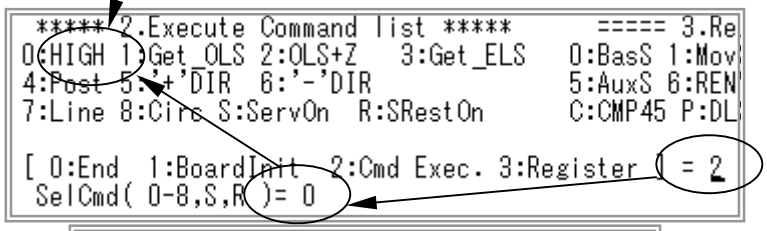
Switches the speed used for operation commands to individual axes.

"CNST" displayed ---- Base speed

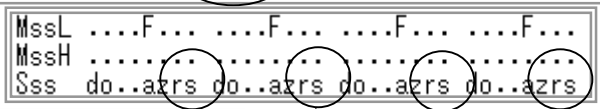
(FL constant-speed start)

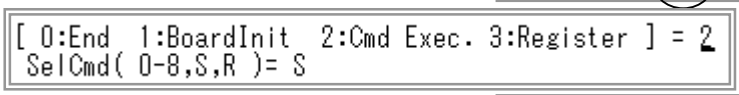
"HIGH"displayed ------Operation speed

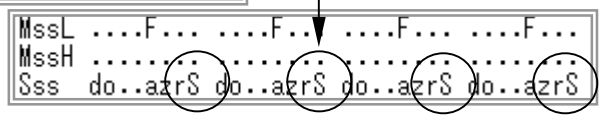
(high-speed start)


***** 2.Execute Command list ***** ===== 3.Re
0:HIGH 1:Get_OLS 2:OLS+Z 3:Get_ELS 0:BasS 1:Mov
4:Post 5:'+'DIR 6:'-'DIR 5:AuxS 6:REN
7:Line 8:Circ S:ServOn R:SRestOn C:CMP45 P:DL
[0:End 1:BoardInit 2:Cmd Exec. 3:Register] = 2
SelCmd(0-8,S,R)= 0

(b) Servo signal output to all axes:







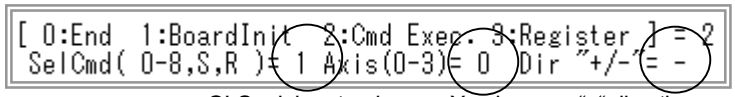
(c) Origin returning operations

There are three origin returning methods provided. If the operation speed is set to "high speed", the Board will perform high-speed origin returning with acceleration and deceleration.

1) OLS Origin Returning (After OLS is detected, get out of and re-enter the origin to complete operation.)

Stops (or decelerates to a stop) when OLS is detected, reverses and gets out of origin, and ends operation when OLS is detected again.

The operation procedure is as follows: set SelCmd=1, specify the desired axis number (0 (X), …), and specify the movement direction (all keys besides the "+" Key specify "–").

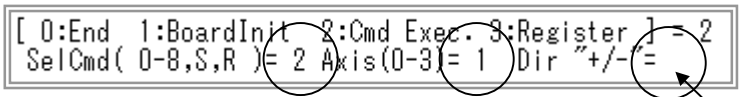


OLS origin returning X-axis "–" direction

2) OLS + Encoder Phase-Z

When OLS is detected (starts decelerating) and ends when phase-Z is detected.

The operation procedure is as follows: set SelCmd=2, specify the desired axis number (0 (X), …), and specify the movement direction (all keys besides the "+" Key specify "–").



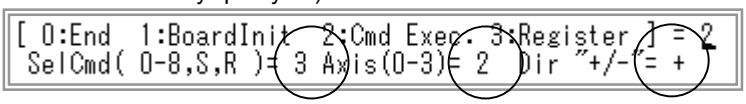
S + phase-Z origin returning Y-axis "–" direction

Press Enter

3) ELS Shared Origin Returning

Stops (or decelerates) when ELS ON signal is detected, reverses and gets out of origin at low speed, and ends operation when ELS OFF signal is detected.

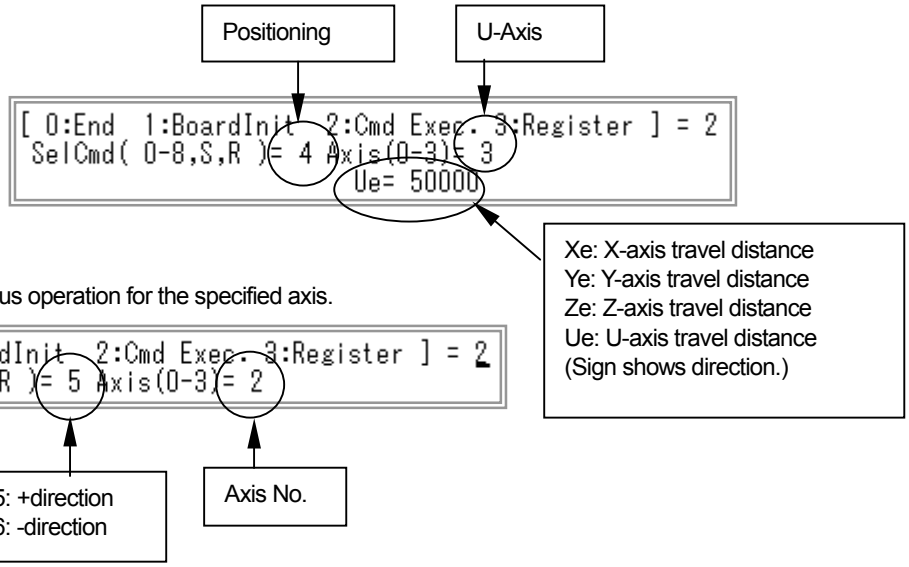
The operation procedure is as follows: set SelCmd=3, specify the desired axis number (0 (X), …), and specify the movement direction (all keys besides the "+" Key specify "–").



ELS dual-use origin returning Z-axis "+" direction

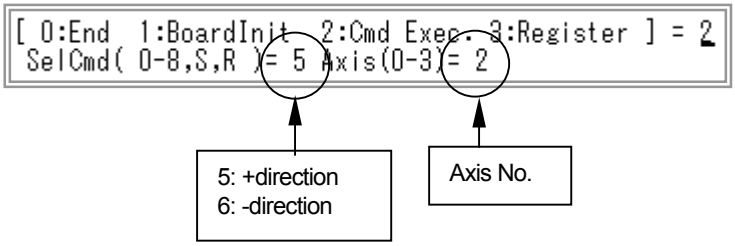
(d) Positioning Operation

Performs 1-axis positioning for the specified axis.



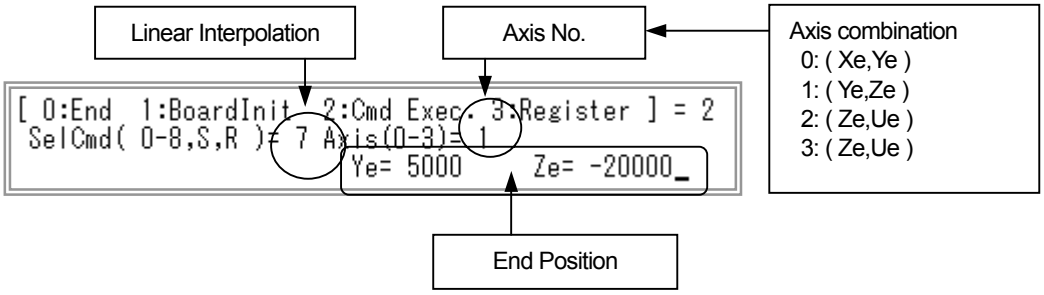
(e) Continuous Operation

Performs 1-axis continuous operation for the specified axis.



(f) Linear Interpolation

Performs linear interpolation operation for the two specified consecutive axes.



(g) Circular Interpolation

Performs circular interpolation operation for the two specified consecutive axes. (See *Linear Interpolation* above for details on specifying the two axes controlled in the operation.)

When the travel distance (end position) for both axes is set to (0, 0), the Board will make a complete circle.